FP-2000 シリーズプリンター
POSPrinter, CashDrawer
OPOS.NET クラスライブラリ
アプリケーションプログラマーズガイド

Ver 1.0

富士通アイソテック株式会社

目次

1．概説 1

1．1．本ドキュメント対象範囲 1
1．2．OPOS.NET クラスライブラリ概要 2
1．3．制限事項 3
1．4．POS プリンターへの接続方法 5

2．インストールについて 7

2．1．インストール手順 7
2．2．アンインストール手順 11
2．3．インストールファイル一覧 14
2．4．設定プログラムの使用法について 15

3．OPOS.NET クラスライブラリ使用方法について 19

3．1．共通 19
3．2．POS プリンター 19
3．3．ドロワー 19
3．4．OPOS.NET クラスライブラリ実装方法 20
3．5．例外 23
エラーコード一覧 24
拡張エラーコード一覧 25

4．OPOS インターフェース仕様（プリンター） 27

4．1．一覧 27
4．2．印字データとエスケープシーケンス 36
4．3．共通プロパティ 43
CapCompareFirmwareVersion プロパティ 43
CapPowerReporting プロパティ 43
CapStatisticsReporting プロパティ 43
CapUpdateStatistics プロパティ 44
CheckHealthText プロパティ 44
Claimed プロパティ 44
DeviceDescription プロパティ 44
DeviceEnabled プロパティ R/W 45
DeviceName プロパティ 46

FreezeEvents プロパティ R/W 46
OutputId プロパティ 46
PowerNotify プロパティ R/W 47
PowerState プロパティ 48
ServiceObjectDescription プロパティ 48
ServiceObjectVersion プロパティ 48
State プロパティ 49
4．4．共通メソッド 50
CheckHealth メソッド 50
Claim メソッド 51
ClearOutput メソッド 52
Close メソッド 52
DirectIO メソッド 53
Open メソッド 63
Release メソッド 63
ResetStatistics メソッド 64
RetrieveStatistics メソッド 64
UpdateFirmware メソッド 65
UpdateStatistics メソッド 66
4．5．専用プロパティ 67
AsyncMode プロパティ R/W 67
CapCharacterSet プロパティ 67
CapCoverSensor プロパティ 68
CapMapCharacterSet プロパティ 68
CapRec2Color プロパティ 68
CapRecBarCode プロパティ 68
CapRecBitmap プロパティ 68
CapRecBold プロパティ 69
CapRecCartridgeSensor プロパティ 69
CapRecColor プロパティ 69
CapRecDhigh プロパティ 69
CapRecDwide プロパティ 69
CapRecDwideDhigh プロパティ 70
CapRecEmptySensor プロパティ 70
CapRecItalic プロパティ 70
CapRecLeft90 プロパティ 70

CapRecMarkFeed プロパティ 70
CapRecNearEndSensor プロパティ 71
CapRecPageMode プロパティ 71
CapRecPapercut プロパティ 71
CapRecPresent プロパティ 71
CapRecRight90 プロパティ 71
CapRecRotate180 プロパティ 72
CapRecStamp プロパティ 72
CapRecUnderline プロパティ 72
CapTransaction プロパティ 72
CartridgeNotify プロパティ R/W 73
CharacterSet プロパティ R/W 74
CharacterSetList プロパティ 75
CoverOpen プロパティ 75
ErrorLevel プロパティ 76
ErrorStation プロパティ 76
ErrorString プロパティ 76
FlagWhenIdle プロパティ R/W 77
FontTypefaceList プロパティ 77
MapCharacterSet プロパティ R/W 77
MapMode プロパティ R/W 78
PageModeArea プロパティ 78
PageModeDescriptor プロパティ 79
PageModeHorizontalPosition プロパティ R/W 80
PageModePrintArea プロパティ R/W 81
PageModePrintDirection プロパティ R/W 82
PageModeStation プロパティ R/W 84
PageModeVerticalPosition プロパティ R/W 84
RecBarCodeRotationList プロパティ 85
RecBitmapRotationList プロパティ 85
RecCartridgeState プロパティ 86
RecCurrentCartridge プロパティ R/W 86
RecEmpty プロパティ 86
RecLetterQuality プロパティ R/W 87
RecLineChars プロパティ R/W 88
RecLineCharsList プロパティ 90

RecLineHeight プロパティ R/W 91
RecLineSpacing プロパティ R/W 93
RecLinesToPaperCut プロパティ 93
RecLineWidth プロパティ 94
RecNearEnd プロパティ 94
RecSidewaysMaxChars プロパティ 95
RecSidewaysMaxLines プロパティ 96
RotateSpecial プロパティ R/W 96
4．6．専用メソッド 98
BeginInsertion メソッド 98
BeginRemoval メソッド 98
ChangePrintSide メソッド 98
ClearPrintArea メソッド 99
CutPaper メソッド 100
EndInsertion メソッド 101
EndRemoval メソッド 101
MarkFeed メソッド 101
PageModePrint メソッド 102
PrintBarCode メソッド 104
PrintBitmap メソッド 117
PrintImmediate メソッド 119
PrintMemoryBitmap メソッド 120
PrintNormal メソッド 122
PrintTwoNormal メソッド 123
RotatePrint メソッド 124
SetBitmap メソッド 127
SetLogo メソッド 129
TransactionPrint メソッド 130
ValidateData メソッド 132
DrawRuledLine メソッド 133
4．7．イベント 134
DirectIOEvent イベント 134
ErrorEvent イベント 135
OutputCompleteEvent イベント 136
StatusUpdateEvent イベント 136
5．OPOS インターフェース仕様（ドロワー） 137

5．1．一覧 137
5．2．共通プロパティ 139
CapCompareFirmwareVersion プロパティ 139
CapPowerReporting プロパティ 139
CapStatisticsReporting プロパティ 139
CapUpdateStatistics プロパティ 140
CheckHealthText プロパティ 140
Claimed プロパティ 140
DeviceDescription プロパティ 140
DeviceEnabled プロパティ R/W 141
DeviceName プロパティ 142
FreezeEvents プロパティ R/W 142
PowerNotify プロパティ R/W 143
PowerState プロパティ 143
ServiceObjectDescription プロパティ 144
ServiceObjectVersion プロパティ 144
State プロパティ 144
5．3．共通メソッド 145
CheckHealth メソッド 145
Claim メソッド 146
Close メソッド 146
DirectIO メソッド 147
Open メソッド 147
Release メソッド 147
ResetStatistics メソッド 148
RetrieveStatistics メソッド 148
UpdateFirmware メソッド 149
UpdateStatistics メソッド 149
5．4．専用プロパティ 150
CapStatus プロパティ 150
CapStatusMultiDrawerDetect プロパティ 150
DrawerOpened プロパティ 150
5．5．専用メソッド 151
OpenDrawer メソッド 151
WaitForDrawerClose メソッド 151
5．6．イベント 152

DirectIOEvent イベント.....152
StatusUpdateEvent イベント 152

6. PosExplorer の外部設定値について.....153

7. ログファイルについて 154

8. プリンター複数台の使用について 155

9. ページモード機能について 156

9.1. バッファリング機能を持つメソッドの排他関係について 156
9.2. ページモードでの印刷データ展開位置 156

修正履歴.....157

1．概説

FP POS プリンターOPOS.NET クラスライブラリおよび、ドロワーOPOS.NET クラスライブラリは、富士通アイソテック株式会社 FP POS プリンター（FP-2200/FP-2100/FP-2000/FP-2000L）、ならびにそれに接続されたドロワーを制御する、OPOS 1.14 POS プリンターデバイス、ドロワーデバイス準拠の OPOS.NET クラスライブラリです。本クラスライブラリの利用にあたっては、「UnifiedPOS Retail Peripheral Architecture」も併せて参照ください。
本書では、「FP」と「FP-2200/FP-2100/FP-2000/FP-2000L」という表現を同じ意味で使用しています。

1．1．本ドキュメント対象範囲

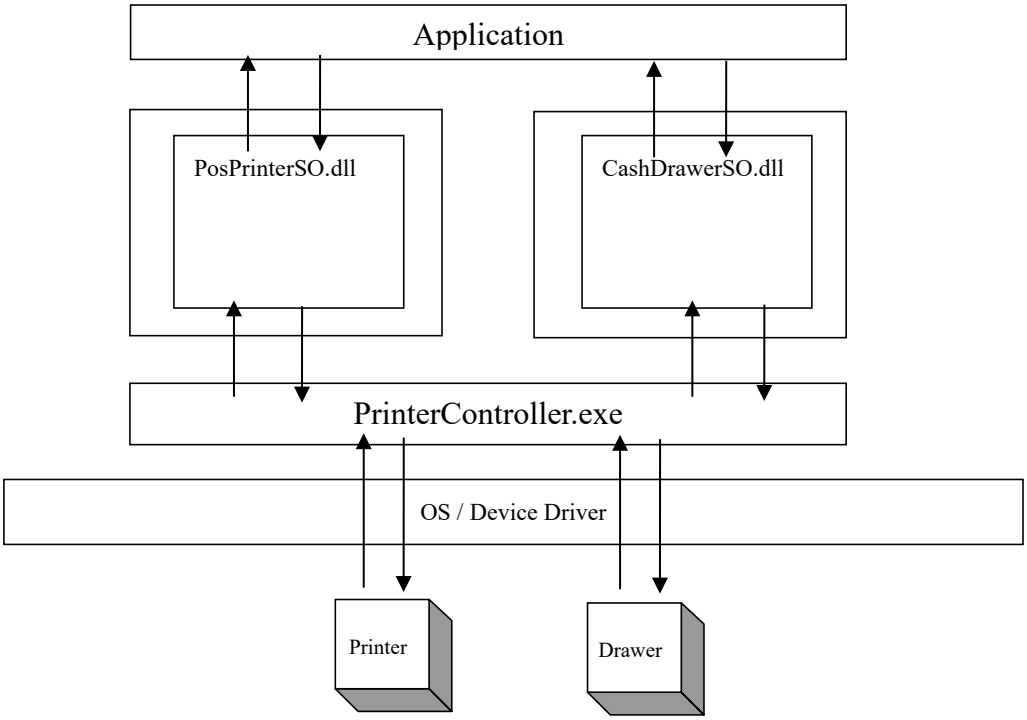
本仕様書（アプリケーションプログラマーズガイド）では、OPOS.NET クラスライブラリを使用するアプリケーション開発者が主に参照することを目的とし、その為に必要な以下の内容を記述しています。

- OPOS.NET クラスライブラリのインストール方法について
- OPOS.NET クラスライブラリの使用方法について
- OPOS.NET クラスライブラリの制限事項
- OPOS.NET クラスライブラリのインターフェース（プロパティ／メソッド／イベント）の説明
- OPOS.NET クラスライブラリの設定項目の説明

1．2．OPOS.NET クラスライブラリ概要

（１）OPOS.NET クラスライブラリ構造図

本クラスライブラリは.NET Framework4.0 にて作成されています。プロパティ、メソッド、イベントをアプリケーションに提供します。アプリケーションは、メソッドの戻り値とパラメータ、プロパティ、イベントを通じて、処理結果を受け取ります。



※本クラスライブラリは、シリアル・USB・LAN インターフェースプリンターの制御をサポートします。各々のプリンターに接続される最大 2 台のドロワーの制御をサポートします。

※複数のインターフェース、複数台のプリンターを設定することが可能です。(「8.プリンター複数台の使用について」を参照」

※LAN インターフェースでのネットワーク接続は 10Base-T, 100Base-TX に対応しています。

※本クラスライブラリはロガーライブラリとして NLog(https://nlog-project.org/)を使用します。

（２）用語

サービスオブジェクト（Service Object：SO）

サービスオブジェクトは、POS for .NET によって定義されるデバイスクラスのインターフェイスを実装するクラスです。プロパティ、およびアプリケーションから呼び出されるメソッドを公開します。

1．3．制限事項

以下の制限事項が存在します。

(1) POS プリンター

【OPOS 仕様上の制限】

OPOS POS プリンターデバイスの全てのインターフェースが提供されますが、以下の制限があります。

① ジャーナル印刷とジャーナルに関するプロパティ設定はサポートしません。

② スリップ印刷とスリップに関するプロパティ設定はサポートしません。

③ イタリック、カスタムカラー、網掛け印刷、カートリッジ機能はサポートしません。

④ レシート印刷の文字フォントの変更はサポートしません。（印字フォント変更）

⑤ 以下のメソッドは、イネーブル後はいずれも必ず Illegal(106)を返します。

PrintTwoNormal メソッド
BeginInsertion メソッド
EndInsertion メソッド
BeginRemoval メソッド
EndRemoval メソッド
ChangePrintSide メソッド
MarkFeed メソッド
ResetStatistics メソッド
RetrieveStatistics メソッド
UpdateStatistics メソッド

【ケーブル抜き挿しによる動作の制限】

イネーブル状態での LAN ケーブル抜き挿しについて、以下の制限があります。

① PC に接続されている LAN ケーブルが抜けた場合の動作はサポートしません。

イネーブル中に LAN ケーブルが抜けた場合、アプリケーションを再起動しても、本クラスライブラリはイネーブルに失敗する可能性があります。

② プリンターに接続されている LAN ケーブルが抜けた場合、その後本クラスライブラリとプリンターの接続が復旧するまでの間にディセーブル動作を行った場合の動作はサポートしません。

上記タイミングでディセーブルを行った場合、アプリケーションを再起動しても、本クラスライブラリはイネーブルに失敗する可能性があります。

印刷中に LAN ケーブルが外れた場合は、LAN ケーブルを接続しプリンターの電源を OFF/ON してください。

(2) ドロワー

【OPOS 仕様上の制限】

OPOS ドロワーデバイスの全てのインターフェースが提供されますが、以下の制限があります。

① PowerNotify プロパティ（電源通知機能設定）

Disabled(0)（通知不可）のみの設定で、変更できません。

② PowerState プロパティ（電源状態）

Unknown(2000)（不明）のみが設定されています。

③ DirectIO メソッド（デバイス固有機能）

サポートしません。イネーブル後は必ず Illegal(106)を返します。

④ WaitForDrawerClose メソッド（ドロワーが閉じられるのを待つ）

サポートしません。イネーブル後は必ず Illegal(106)を返します。

⑤ DirectIOEvent イベント（デバイス固有イベント）

サポートしません。

⑥ DrawerOpened プロパティ、StatusUpdateEvent イベント

ドロワーのステータス通知は、CapStatus プロパティが TRUE で且つ、ドロワーが接続されているプリンターに対して、プリンタークラスライブラリがイネーブル(DeviceEnabled=TRUE)されている必要があります。これらの条件が満たされない場合、ドロワーステータスは通知されません。

【ドロワーハードウエア仕様上の制限】

・ドロワーの電源状態の通知機能をサポートしません。

(3) POS プリンター、ドロワー共通の制限事項

本クラスライブラリはスレッドセーフではありません。異なるスレッドからメソッド、プロパティにアクセスした場合、予期せぬ結果が発生する場合があります。マルチスレッド環境化では、プロパティのアクセスやメソッドの実行はクリティカルセクション等の排他処理を実装して、同時にメソッド、プロパティが実行されないようにしてください。

(4) Windows ドライバと本クラスライブラリを同一システムにインストールした場合の制限

Windows ドライバと本クラスライブラリを同一システムに両方ともインストールした場合、本クラスライブラリから正常に印刷できない等の問題が発生します。

このような場合は、Windows ドライバをアンインストールし、使用しないことを推奨します。

(5) 本クラスライブラリでは、以下のプリンター機能についてサポートしておりません。

・「エラー時のリトライ印刷」

・「プラグアンドプレイ」

※ 上記設定はセットアップツールでの設定時に”無効”に変更されます。

(6) OPOS ドライバ(OCX)との同時使用による動作はサポートしておりません。

(7) デバイスオープン状態での PC スリープまたは休止モード復帰による動作はサポートしておりません。

1．4．POS プリンターへの接続方法

POS プリンターの設定は以下の設定(編みかけ部分)で動作するようにしてください。その他の値は、PosExplorer 外部設定値による設定やインストーラに付属する設定プログラムにて設定可能な値です。

メモリスイッチ 1

No.	設定項目	詳細設定項目
1	電源 ON 通知	※PosExplorer 外部設定値で設定
2	受信バッファ	64K バイト
3	Busy 条件	バッファフル
4	受信エラー処理	印字
5	自動改行	常時無効
6	DSR(#6)リセット	無効
7	INT(#25)リセット	無効
8	USB ソフトリセット	有効

メモリスイッチ 2

No.	設定項目	詳細設定項目
1	カバーオープン	自動復帰エラー
2	エラー復帰	コマンド
3	印刷モード (COM IF)	逐次
4	印刷モード (BT IF)	逐次
5	印刷モード (Other IF)	逐次
6	シリアル番号	有効
7	ASB	有効
8	フォント B	モード 1
9	韓国フォント	※設定プログラムから設定

プリント

No.	設定項目	詳細設定項目
1	DPI	※設定プログラムから設定
2	印字幅	※設定プログラムから設定
3	印刷速度	※設定プログラムから設定
4	印刷濃度	※設定プログラムから設定
5	エラー時のリトライ印刷	無効
7	言語選択	※設定プログラムから設定

ハードウェア

No.	設定項目	詳細設定項目
1	メモリ（ユーザ）	192KB
2	メモリ（グラフィック）	384KB (FP-2000) 896KB (FP-2200/FP-2100/FP-2000L)
3	カバー閉時のカット	※設定プログラムから設定
4	カッタ動作	※設定プログラムから設定
5	PNE 検出	※設定プログラムから設定

ブザー

No.	設定項目	詳細設定項目
1	エラー通知ブザー	※設定プログラムから設定
2	印刷時のブザー音色	※設定プログラムから設定
3	印刷時のブザー回数	※設定プログラムから設定
4	カット後のブザー	※設定プログラムから設定

インターフェース

No.	設定項目	詳細設定項目
1	プロトコル	XON/XOFF
2	USB	プリンタ
3	プラグアンドプレイ	無効

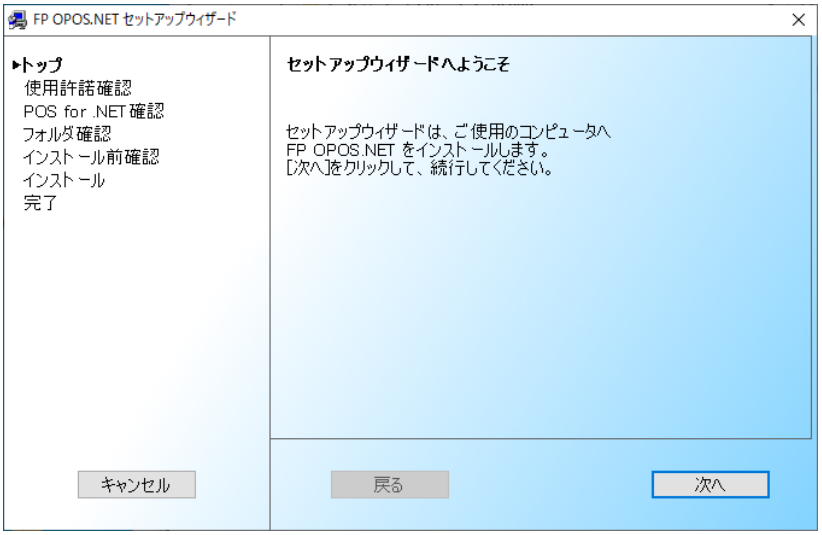
2. インストールについて

本クラスライブラリのインストールは下記手順で行います。

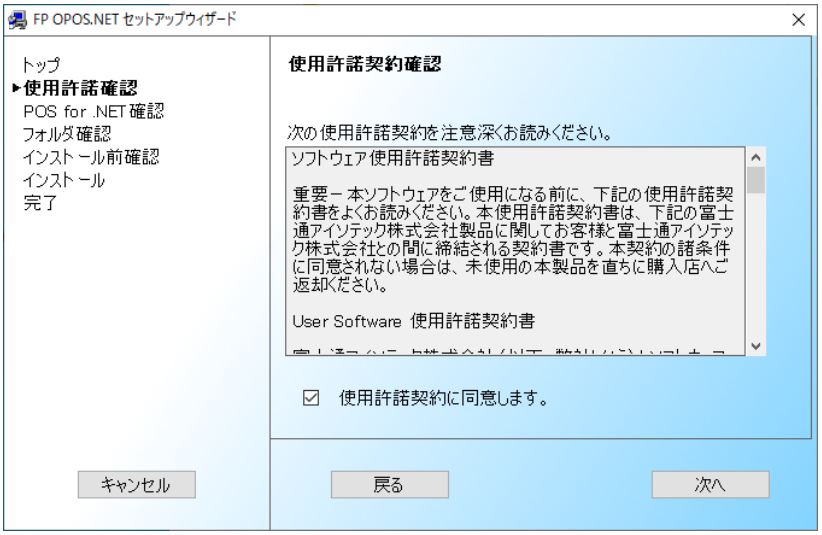
2. 1. インストール手順

1. OPOS.NET フォルダ内にある、”Setup.exe”をダブルクリックします。
   [ユーザーアカウント制御]ダイアログが表示された場合は、[はい]をクリックします。

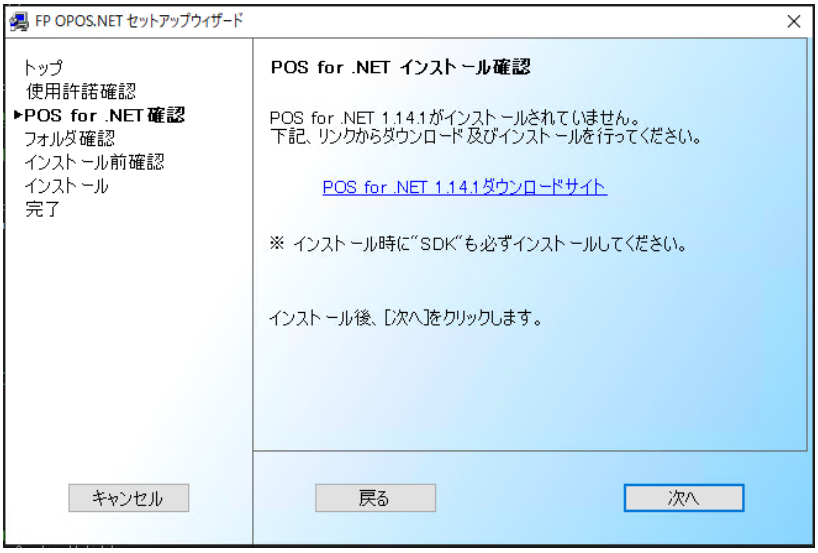
2. セットアップウィザード画面が表示されます。[次へ]をクリックします。



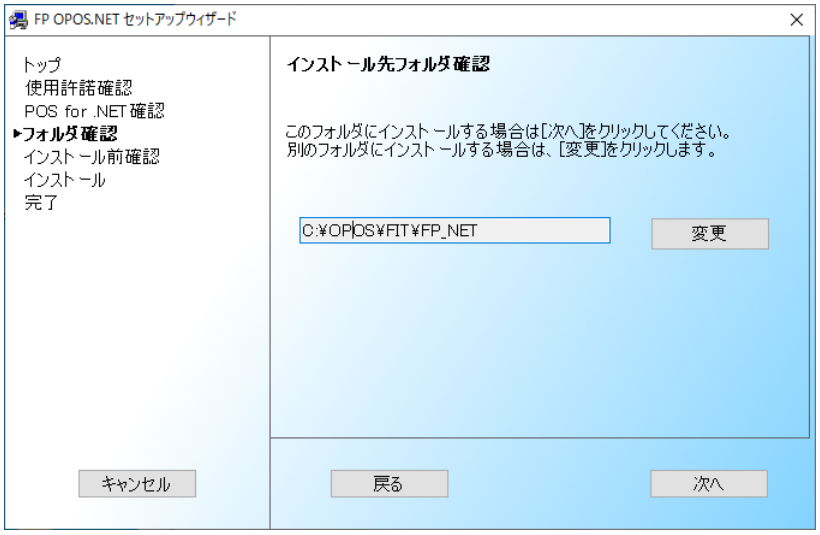
3. 使用許諾契約内容を注意深く読み、”使用許諾契約に同意します”のチェックを入れ、
   [次へ]をクリックします。



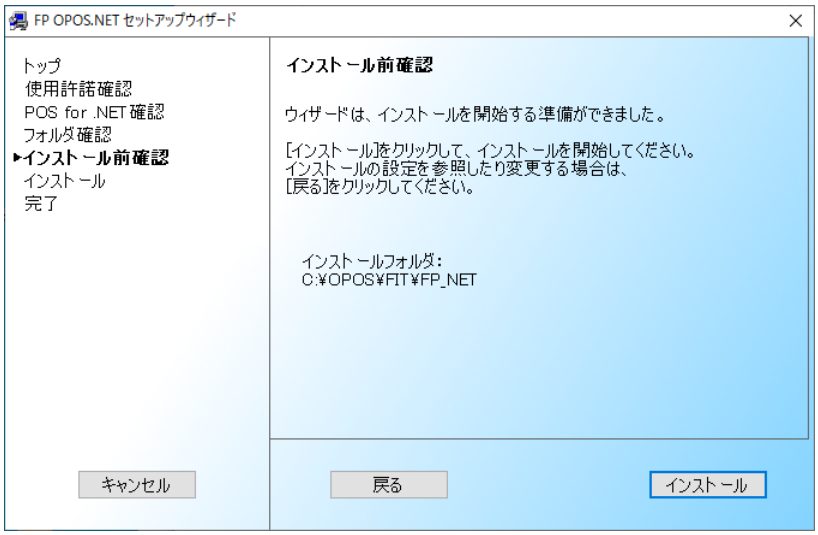
4. POS for .NET 1.14.1 がインストールされていない場合は、下記画面のリンクをクリックし、POS for .NET 1.14.1 のダウンロードとインストールを行います。
インストール完了後、[次へ]をクリックします。
※ POS for .NET 1.14.1 がインストール済だった場合は、本画面は表示されませんので、5 の手順に移動します。



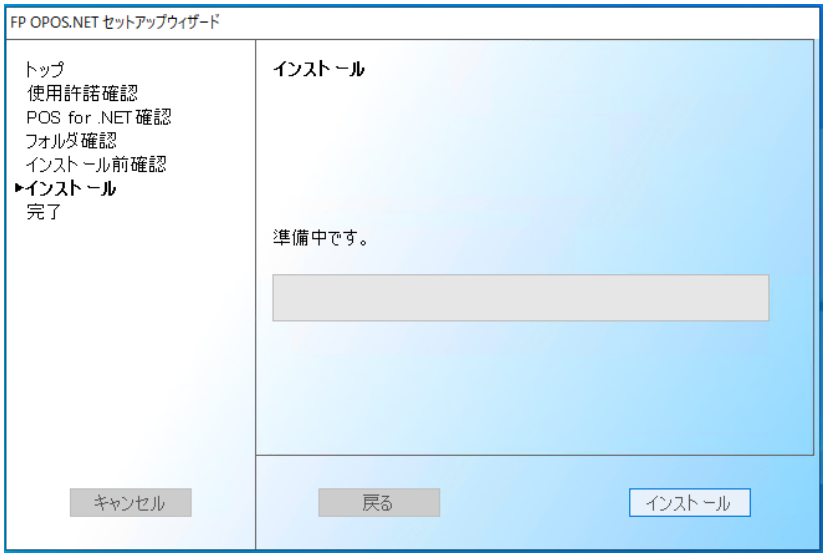
5. OPOS.NET ドライバをインストールするフォルダを指定します。
必要に応じてフォルダを変更後、[次へ]をクリックします。
※ ドライブ直下や、サーバーへのインストールはできません。



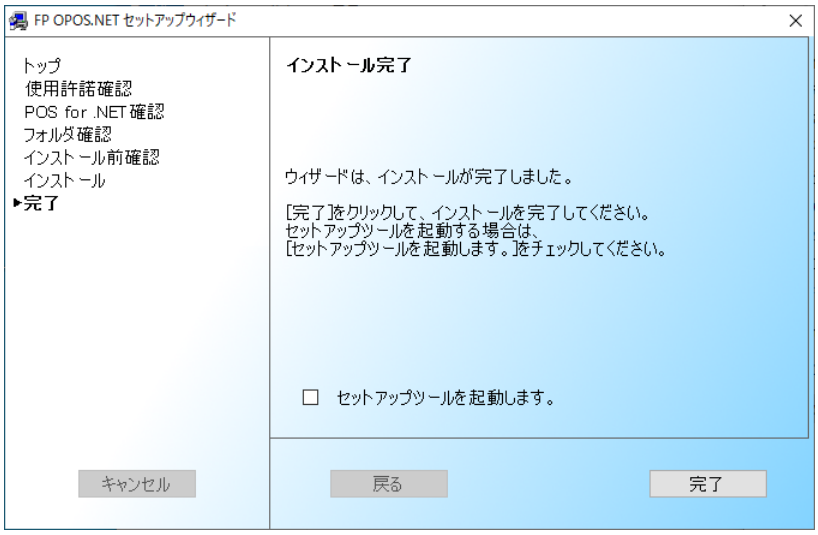
6. インストール内容を確認し、[インストール]をクリックします。



7. インストール中は下記の画面が表示されますので、しばらくお待ちください。



8. インストールが完了しました。
必要に応じて"セットアップツールを起動します"にチェックを付けて、[完了]をクリックし、インストーラを終了します。



2．2．アンインストール手順

1.　次の手順で、アンインストール画面を表示します。

【Windows 10(ver1703~) / Windows Server 2019 の場合】

・デスクトップ画面左下の Windows ロゴがついたボタンをクリックします。

・アプリケーションの一覧から[Windows システムツール]→[コントロールパネル]→[プログラム]→[プログラムと機能]をクリックします。

【Windows 10(~ver1607) / Windows Server 2016 の場合】

デスクトップ画面左下の Windows ロゴがついたボタンを右クリック→[プログラムと機能]を選択します。

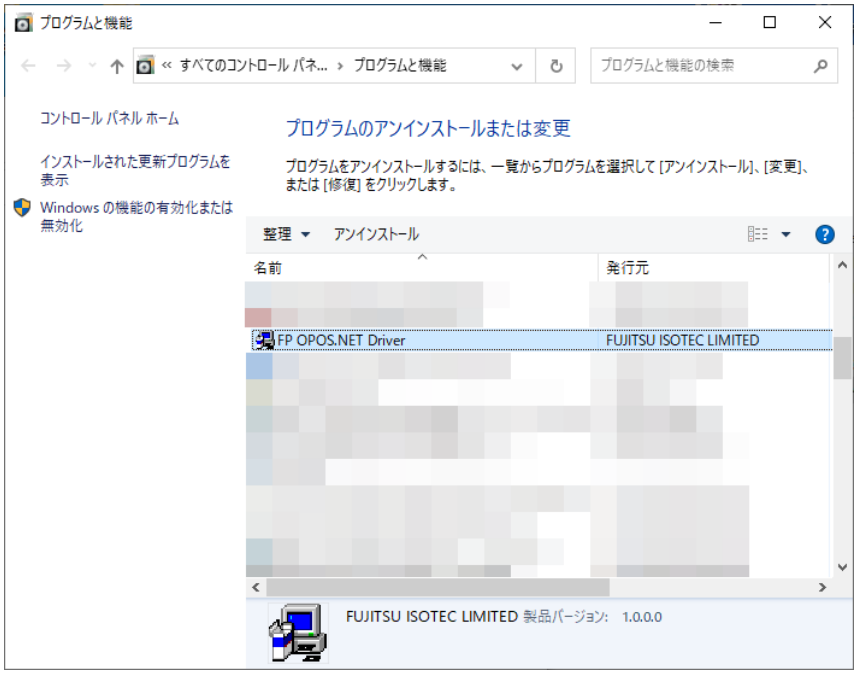
【Windows 8.1 / Windows Server 2012R2 の場合】

Windows ロゴキー+[X]キーを押し、画面の左下に表示されるメニューから、「プログラムと機能」を選択してください。

【Windows 7 の場合】

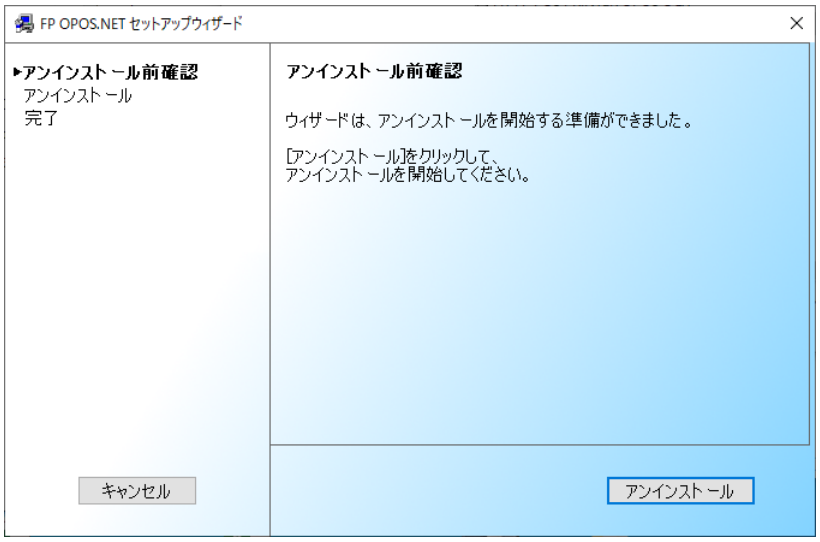
・デスクトップ画面左下の Windows ロゴがついたボタンをクリックします。

・[コントロールパネル]→[プログラムのアンインストール]をクリックします。

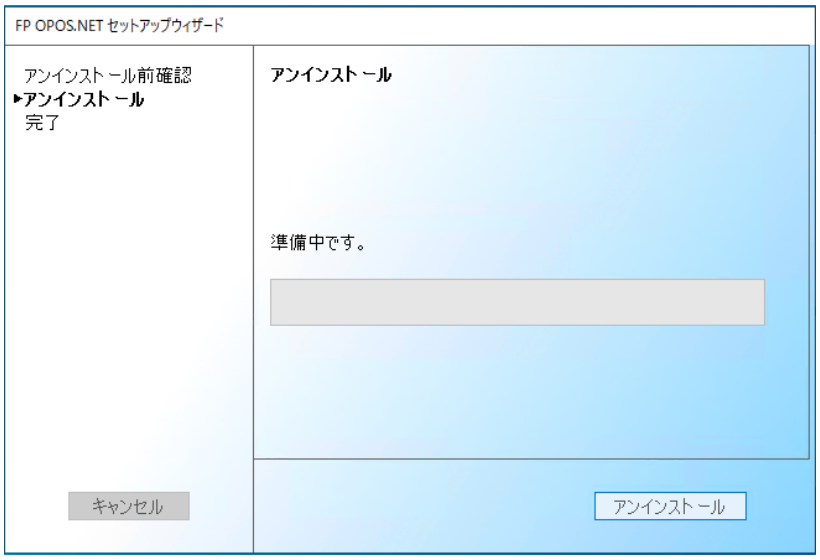


2. “FP OPOS.NET Driver”を選択し、“アンインストール”をクリックします。

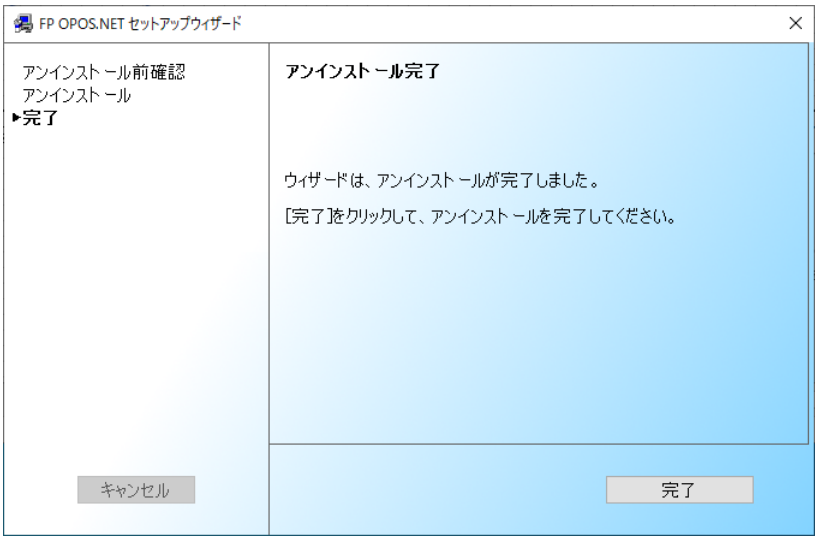
3. アンインストール確認画面が表示されます。[アンインストール]をクリックします。



4. アンインストールが実行されます。しばらくお待ちください。



5. アンインストールが完了しました。
[完了]をクリックして、アンインストーラを終了します。



6. インストールしたフォルダ（[システムドライブ]:¥ OPOS¥FIT¥FP_NET）に、アンインストーラが削除できなかったファイルが残っている場合がありますので、これらは手動で削除してください。

以上でアンインストールは終了です。

2．3．インストールファイル一覧

本クラスライブラリのファイル群は以下のように配置されます。

[システムドライブ]:¥OPOS¥FIT¥FP_NET

BitImage.dll	PrinterSO 依存ファイル。Bitmap プリント用
BitImage_win32.dll	
CashDrawer.dll	ドロワーのサービスオブジェクト
NLog.dll	ロガーファイルライブラリ用オブジェクト
PosPrinter.dll	POS プリンターのサービスオブジェクト
PrintController.exe	プリンター制御モジュール
SharedObject.dll	通信用オブジェクト
Setup.exe	本クラスライブラリのインストーラ
Setup.xml	本クラスライブラリのインストーラ
UIData.xml	本クラスライブラリのインストーラ

[システムドライブ]:¥OPOS¥FIT¥FP_NET¥ftfpcp

Win32¥ftfpcp.dll
WIn64¥ftfpcp.dll

[システムドライブ]:¥OPOS¥FIT¥FP_NET¥tdbm

Win32¥tdbm.dll
WIn64¥tdbm.dll

[システムドライブ]:¥OPOS¥FIT¥FP_NET¥APG

本 APG ファイル

[システムドライブ]:¥OPOS¥FIT¥FP_NET¥SampleProgram

Object： サンプルプログラムの実行ファイル
Source： サンプルプログラムソース

[システムドライブ]:¥OPOS¥FIT¥FP_NET¥SetupTool

本クラスライブラリのセットアップツール

2．4．設定プログラムの使用法について

本ドライバを使用するためのプリンター情報の登録と、プリンターの設定を行います。
設定を行う際は、必ず設定するプリンターだけを接続してください。(複数台接続している場合は、設定できません。)

1.　実行

OS 別に以下の操作を行い、設定プログラムを起動します。
※[ユーザーアカウント制御]ダイアログが表示された場合には、[はい]をクリックしてください。

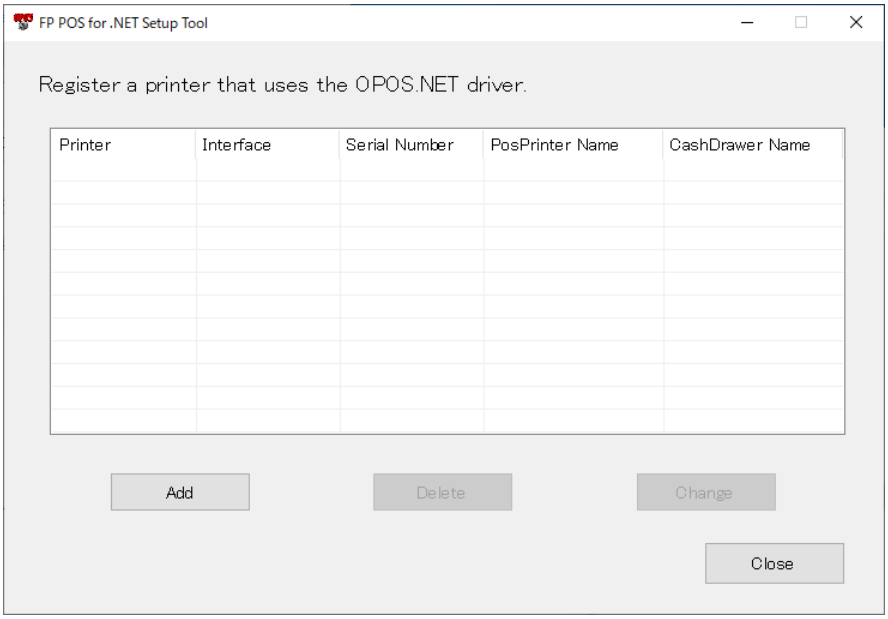
【Windows 10(ver1607～) / Windows Server 2016 以降の場合】
① デスクトップ画面左下の Windows ロゴがついたボタンをクリックします。
② [FIT FP Series Printer] →[OPOS.NET Setup Tool]をクリックしてください。

【Windows 10(～ver1511) の場合】
① デスクトップ画面左下の Windows ロゴがついたボタンをクリックします。
② [すべてのアプリ]→[FIT FP Series Printer] →[OPOS.NET Setup Tool]をクリックしてください。

【Windows 8.1 / Windows Server 2012 R2 の場合】
① スタート画面でマウスを動かし をクリックします。
② アプリ一覧で[OPOS.NET Setup Tool]タイルをクリックしてください。

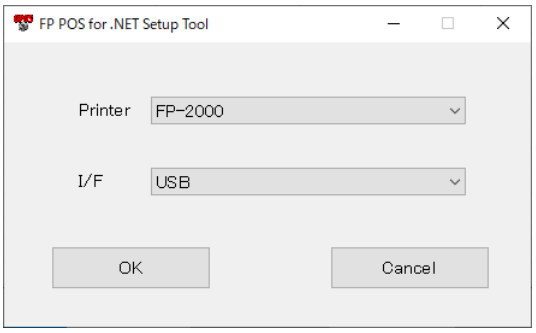
【Windows 7 の場合】
① デスクトップ画面左下の Windows ロゴがついたボタンをクリックします。
② [すべてのプログラム]→[FIT FP Series Printer]→ [OPOS.NET] →[OPOS.NET Setup Tool] をクリックしてください。

2.　登録済みプリンター情報の確認



プリンター情報を新たに追加する場合は、[Add]をクリックします。
削除する場合は、登録リストから削除するプリンターを選択して、[Delete]をクリックします。
プリンター情報を変更する場合は、登録リストから変更するプリンターを選択して、[Change]をクリックします。

3.　プリンター/ポート選択（新規追加）

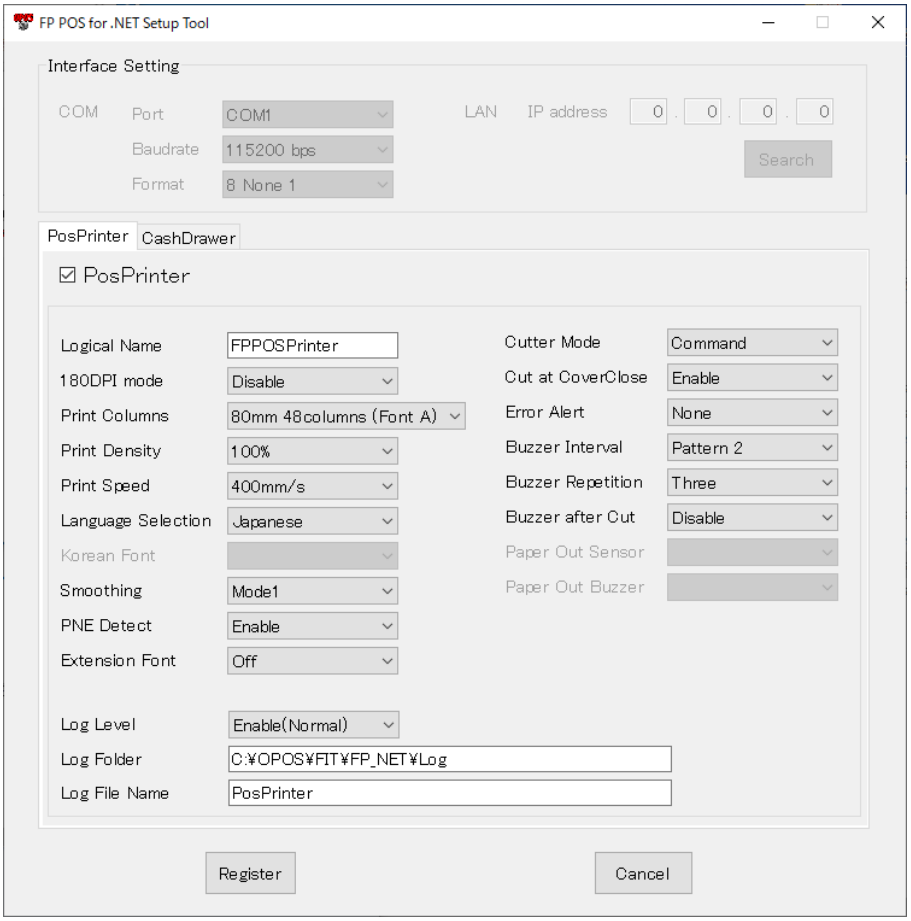


追加するプリンターとインタフェースを選択し、[OK]をクリックします。
なお、登録できる台数はインタフェースによって異なります。

USB/COM：2 台まで
LAN：255 台まで

4. PosPrinter/CashDrawer 詳細設定

（画面は FP-2200 の例です）



使用するインタフェースの設定及び、PosPrinter/CashDrawer の設定を行います。
PosPrinter 及び CashDrawer で、追加するものにチェックを入れ、プリンターの設定変更を行ってください。

[Register]クリック時にエラーが発生した場合、エラー要因を確認して、再度設定を行ってください。

＜エラー要因＞

・変更するプリンター以外のプリンターが接続されている。
・ケーブルが接続されていない。
・プリンターの電源が入っていない。
・カバーが開いている。
・用紙がない。
・既に他でポートが使用されている。
・通信条件がプリンターと一致していない。（シリアル接続時）
・IP アドレスに間違いがある。（LAN 接続時）

設定内容一覧

※　プリンターによって、設定可能な項目が異なります。

設定項目			説明
Interface Setting			使用するインタフェースの設定を行います。
	COM	Port	ポート番号を指定します。
		Baudrate	通信速度を指定します。 プリンター設定と同じボーレートに設定する必要があります。
		Format	通信フォーマットを指定します。 プリンター設定と同じフォーマットに設定する必要があります。
	LAN	IP address	プリンターの IP アドレスを設定します。 [Search]からプリンターを検索して指定することも可能です。
PosPrinter			PosPrinter を登録する場合に、チェックを入れます。
	Logical Name		PosPrinter の論理デバイス名を指定します。
	180DPI mode		DPI モードを設定します。 Enable を選択すると、180DPI モードが有効となります。
	Print Columns		用紙の横幅と 1 行に印字可能な文字数を指定します。
	Print Density		印字濃度を指定します。
	Print Speed		印刷速度を指定します。 プリンターによって、印刷速度が異なります。
	Language Selection		印刷するプリンターの言語仕様を指定します。
	Korean Font		韓国語フォントの書体を指定します。
	Smoothing		スムージング機能を指定します。 フォント印字時に 2 倍角印字以上が指定された場合や、RecLetterQuality プロパティを TRUE に設定した場合に、スムージング処理を行います。
	PNE Detect		ペーパーニアエンドの通知を指定します。
	Extension Font		拡張フォントを指定します。
	Cutter Mode		用紙カットのモードを指定します。
	Cut at Cover Close		カバーと閉じた時のカット動作を指定します。
	Error Alert		エラー発生時のブザー鳴動を指定します。
	Buzzer Interval		ブザーの鳴動間隔パターンを指定します。
	Buzzer Repetition		ブザーの鳴動回数を指定します。
	Buzzer after Cut		用紙カット後のブザー鳴動を指定します。
	Paper Out Sensor		排紙センサー機能を指定します。
	Paper Out Buzzer		用紙抜取待ち時のブザー鳴動を指定します。
	Log Level Log Folder Log File Name		本ドライバのログ機能を指定します。 詳細は"7. ログファイルについて"をご参照ください。
CashDrawer			CashDrawer を登録する場合に、チェックを入れます。
	Logical Name		CashDrawer の論理デバイス名を指定します。
	Drawer No		ドロワー番号を指定します。
	Drawer Status		ドロワーステータスを指定します。 ドロワーオープン・クローズのイベントが逆転する場合に設定します。
	Log Level Log Folder Log File Name		本ドライバのログ機能を指定します。 詳細は"7. ログファイルについて"をご参照ください。

3．OPOS.NET クラスライブラリ使用方法について

3．1．共通

アプリケーションは以下の手順にて本クラスライブラリを使用します。

① **Open** メソッド：デバイスをオープンするときに Open を呼び出します。
② **Claim** メソッド：デバイスに排他アクセスできるようにするために呼び出します。
排他使用するデバイスでは必須であり、共有可能デバイスには任意のメソッドです。
③ **DeviceEnabled** プロパティ：デバイスを動作させる場合 **TRUE** に設定します。
④ デバイスを使用します。（各プロパティ・メソッド・イベント）
⑤ **DeviceEnabled** プロパティ：**FALSE** に設定してデバイスをディセーブルにします。
⑥ **Release** メソッド：デバイスの排他アクセスを解除するために呼び出します。
⑦ **Close** メソッド：デバイスとそのリソースを解放するときに Close を呼び出します。

その他使用方法の詳細については、「UnifiedPOS Retail Peripheral Architecture」のドキュメントを参照してください。

3．2．POS プリンター

POS プリンターは「レシート」のみをサポートします。従って、それ以外（ジャーナル/スリップ）のメソッド・プロパティは、インターフェースは提供されますが、動作はサポートされません。
POS プリンターは一般的な出力モデルに従い、同期/非同期の出力が可能です。
また、POS プリンターは排他的に使用するデバイスです。

3．3．ドロワー

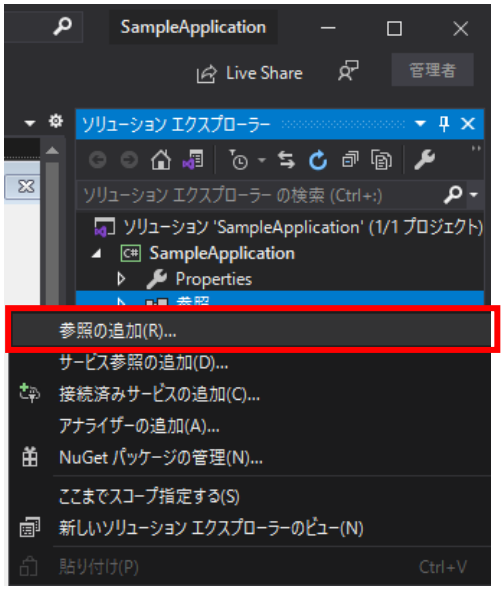
ドロワーも使用方法については、POS プリンターと同様ですが、Claim メソッドを実行しなくともすべての機能が実行可能です。ただし、あるアプリケーションが Claim メソッドにより排他権を取得しているとき、同じ名前でイネーブルしたアプリケーションは、OpenDrawer メソッドを実行できません。排他権を獲得しているアプリケーションが存在しない場合はこの限りではありません。

3．4．OPOS.NET クラスライブラリ実装方法

本クラスライブラリをアプリケーションに実装する方法を以下に記します。

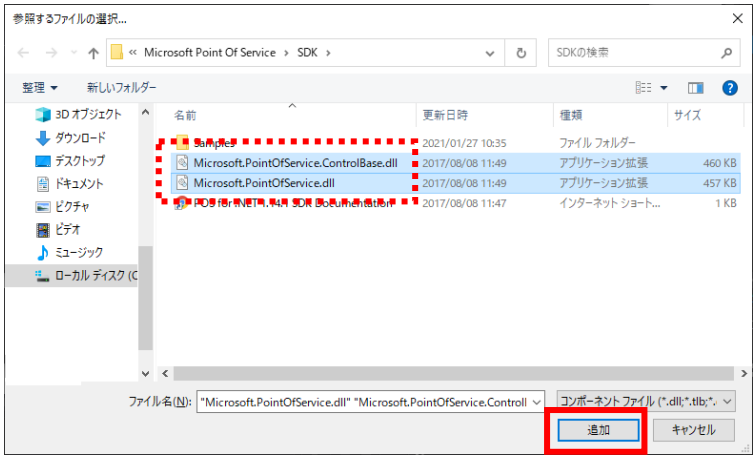
Visual Studio 2019 で実装する場合

1. **Visual Studio 2019** を起動し、作成するプロジェクト種類を選択します。ここでは、[Windows フォーム アプリケーション（.NET Framework）] を選択した例を示します。選択後、[ソリューション エクスプローラー]の"参照"を右クリックし、[参照の追加...]を選択します。

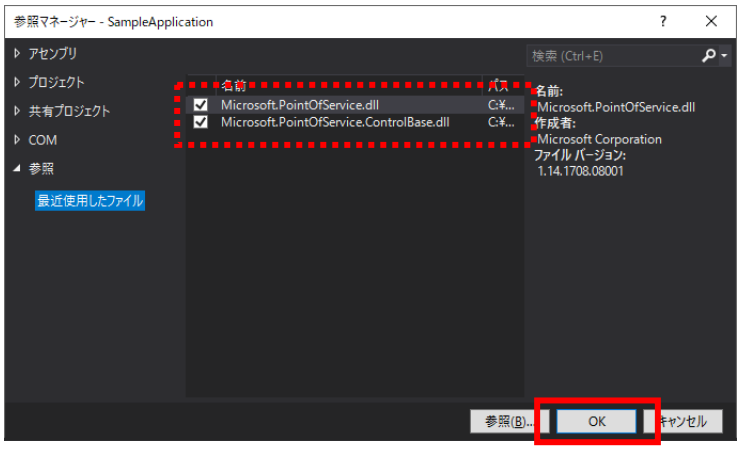


2. 参照マネージャーが表示されるので[参照…]を選択します。

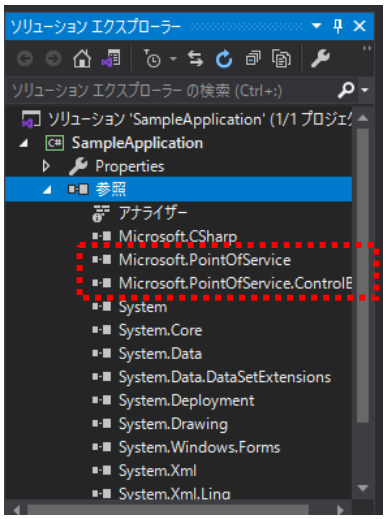
3. “C:¥Program Files (x86)¥Microsoft Point Of Service¥SDK”(※1) にある下記のファイルを選択し、[追加]をクリックします。
・Microsoft.PointOfService.ControlBase.dll (※2)
・Microsoft.PointOfService.dll (※2)
※1：32 ビットの場合は” C:¥ Program Files¥Microsoft Point Of Service¥SDK”
※2： ファイルがない場合は、再度 POS for .NET のインストールを行い、”SDK”を追加してください。



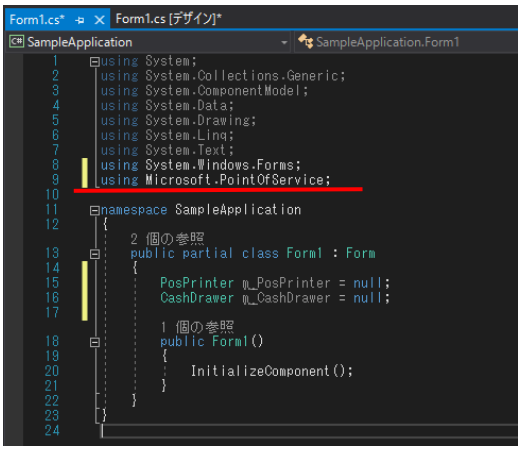
4. 参照マネージャーに追加したファイルがあることを確認し、チェックをつけ[OK]をクリックします。



5. [ソリューション エクスプローラー]の"参照"を開き、”Microsoft.PointOfService” 及び、Microsoft.PointOfService.ControlBase”が追加されていることを確認します。



6. ソースコード内に”using Microsoft.PointOfService”を追加し、ご使用ください。



3．5．例外

プロパティ DeviceControlVersion、DeviceControlDescription、および State へのアクセスを除き、すべての POS for .NET メソッドの呼び出しやプロパティアクセスは失敗時に PosControlException をスローする可能性があります。

名前	説明
ErrorCode	例外の原因を表すエラーコード。エラーコード一覧を参照
ErrorCodeExtended	例外の原因を表す拡張エラーコード。これには、サービス固有の値が格納されることがあります。

エラーコード一覧

説明

このプロパティは各メソッドが設定します。プロパティを取得したときや書込み可能プロパティを設定したときも設定されます。

このプロパティはいつでも読取り可能です。**Open** メソッドを呼び出すまでは、Closed (101)を戻します。結果コードの値は次のとおりです。

値	意味
Closed (101)	クローズされているデバイスにアクセスしようとしました。
NotClaimed (103)	メソッドまたはプロパティ設定処理を使用する前に、排他アクセス権の獲得をしなければならない排他使用デバイスにアクセスしようとしました。
NoService (104)	コントロールがサービスオブジェクトと通信できません。おそらく、セットアップエラーかコンフィギュレーションエラーを修正しなければなりません。
Disabled (105)	デバイスをディセーブルしているときには動作を実行できません。
Illegal (106)	デバイスに無効な動作か、サポートされていない動作を実行しようとしたか、無効なパラメータ値を使用しました。
NoHardware (107)	POS プリンターがオフ、またはオフラインです。
Failure (111)	デバイスがシステムに接続され、電源が入っていて、オンラインですが、リクエストされた処理をデバイスが実行できません。
Timeout (112)	デバイスからの応答を待ち合わせていたサービスオブジェクトがタイムアウトしたか、サービスオブジェクトからの応答を待ち合わせていたコントロールがタイムアウトしました。
Busy (113)	現在の SO の状態は、この要求を受け付けられません。例えば、非同期出力が実行中の場合、いくつかのメソッドは受け付けられません。
Extended (114)	固有エラー状態が発生しました。ErrorCodeExtended プロパティでエラー状態コードを確認できます。

拡張エラーコード一覧

説明

ErrorCode が Extended (114)の場合、このプロパティにデバイスクラスの説明で記述されているクラス固有のエラー情報値が設定されます。

ErrorCode が別の値の場合、サービスオブジェクトが SO 固有の値をこのプロパティに設定できます。これらの値に意味があるのは、アプリケーションが SO 固有の値を追加して処理する場合だけです。

本クラスライブラリでは、下記の値をとります。

ErrorCodeExtended	定数名（内容）	詳細
201	ExtendedErrorCoverOpen	プリンターカバーが開いています。 エラーを復旧するには、カバーを閉じてください。このとき、本クラスライブラリをディセーブル、**Release**,**Close** する必要はありません。印字途中でカバーオープンが発生した場合、カバークローズ後は未印字データが印字されます。ただし、印字データが途切れて印字される可能性があります。
203	ExtendedErrorReceiptEmpty	レシート切れが発生しました。エラーを復旧するには、レシート紙を給紙してください。このとき、本クラスライブラリをディセーブル、**Release**,**Close** する必要はありません。印字途中でレシート切れが発生した場合、レシート給紙後は未印字データが印字されます。
206	ExtendedErrorTooBig	ビットマップが大きすぎて、プリンターが処理できません ビットマップファイルを確認してください。
207	ExtendedErrorBadFormat	ビットマップファイルのフォーマットとして正しくありません。 ファイル名が間違っているか、ファイルの内容が壊れている可能性があります。
281	BadFile	ファームウェアファイルのフォーマットとして正しくありません ファイル名が間違っているか、ファイルの内容が壊れている可能性があります。

ErrorCodeExtended	定数名（内容）	詳細
10001	Blackmark	プリンターからブラックマークセンサエラーが発生しました。用紙がブラックマーク対応でない可能性があります。このとき、本クラスライブラリをディセーブル、**Release**,**Close** する必要はありません。レシートをブラックマーク対応のものに変更してください。※設定プログラムでブラックマークエラー検知をオン(Black Mark を Enable にチェック)にしていない場合は発生しません。
10003	Fatal	プリンターにて致命的エラーが発生しました。復旧方法はありませんので、本クラスライブラリを **Close** してください。
10006	Overheat	プリンターヘッド昇温エラーが発生しました。エラーを復旧するには、ヘッド温度が低下するのを待って、再びエラーが発生しなくなるまで時間をおいてください。このとき、本クラスライブラリをディセーブル、**Release**,**Close** する必要はありません。印字途中でヘッド昇温が発生した場合、ヘッド温度回復後は未印字データが印字されます。ただし、印字データが途切れて印字される可能性があります。
10008	Cutterjam	カッタージャムエラーが発生しました。エラーを復旧するには、カバーを開き、原因（紙詰まりなど）を除去後、カバーを閉じてください。 カバーが開かない場合は、プリンターのユーザーズマニュアルの手順に従ってください。

4. OPOS インターフェース仕様（プリンター）

4．1．一覧

プロパティ

共通	型	ｱｸｾｽ	初期化	初期値･書込み条件
CapCompareFirmwareVersion	bool	R	Open	TRUE
CapPowerReporting	PowerReporting	R	Open	Standard (1)
CapStatisticsReporting	bool	R	Open	FALSE
CapUpdateStatistics	bool	R	Open	FALSE
CheckHealthText	string	R	Open	“”
Claimed	bool	R	Open	FALSE
DeviceEnabled	bool	R/W	Open&Claim	FALSE
FreezeEvents	bool	R/W	Open	FALSE Open 後、書込み可能
OutputId	int	R	Open	1
PowerNotify	PowerNotification	R/W	Open	Disabled (0) Open 後、書込み可能。 Enabled 後、書込み不可
PowerState	PowerState	R	Open	Unknown (2000)
State	ControlState	R	--	Closed (1)
ServiceObjectDescription	string	R	Open	“Fujitsu Isotec FP POS Printer Service Object”
ServiceObjectVersion	Version	R	Open	1013XXX
DeviceDescription	string	R	Open	“FP 1 Station Thermal POSPrinter (C)20xx Fujitsu Isotec Limited”
DeviceName	string	R	Open	“FP 1 Station Thermal POSPrinter”

専用	型	ｱｸｾｽ	初期化	初期値と備考
CapCharacterSet	CharacterSetCapability	R	Open	Kanji (11)
CapConcurrentJrnRec	bool	R	Open	FALSE
CapConcurrentJrnSlp	bool	R	Open	FALSE
CapConcurrentPageMode	bool	R	Open	FALSE
CapConcurrentRecSlp	bool	R	Open	FALSE
CapCoverSensor	bool	R	Open	TRUE
CapMapCharacterSet	bool	R	Open	TRUE
CapTransaction	bool	R	Open	TRUE
CapJrnPresent	bool	R	Open	FALSE
CapJrn2Color	bool	R	Open	FALSE
CapJrnBold	bool	R	Open	FALSE
CapJrnDhigh	bool	R	Open	FALSE
CapJrnDwide	bool	R	Open	FALSE
CapJrnDwideDhigh	bool	R	Open	FALSE
CapJrnEmptySensor	bool	R	Open	FALSE
CapJrnItalic	bool	R	Open	FALSE
CapJrnNearEndSensor	bool	R	Open	FALSE
CapJrnUnderline	bool	R	Open	FALSE
CapJrnCartridgeSensor	PrinterCartridgeSensors	R	Open	None(0)
CapJrnColor	PrinterColors	R	Open	None(0)
CapRecPresent	bool	R	Open	TRUE
CapRec2Color	bool	R	Open	初期値は PosExplorer 外部設定値によって変化
CapRecBarCode	bool	R	Open	TRUE
CapRecBitmap	bool	R	Open	TRUE
CapRecBold	bool	R	Open	TRUE
CapRecDhigh	bool	R	Open	TRUE
CapRecDwide	bool	R	Open	TRUE
CapRecDwideDhigh	bool	R	Open	TRUE
CapRecEmptySensor	bool	R	Open	TRUE
CapRecItalic	bool	R	Open	FALSE
CapRecLeft90	bool	R	Open	TRUE
CapRecNearEndSensor	bool	R	Open	初期値は PosExplorer 外部設定値によって変化
CapRecPapercut	bool	R	Open	TRUE
CapRecRight90	bool	R	Open	TRUE
CapRecRotate180	bool	R	Open	TRUE
CapRecStamp	bool	R	Open	FALSE
CapRecUnderline	bool	R	Open	TRUE
CapRecCartridgeSensor	PrinterCartridgeSensors	R	Open	None(0)

専用	型	ｱｸｾｽ	初期化	初期値と備考
CapRecColor	PrinterColors	R	Open	None(0)
CapRecMarkFeed	PrinterMarkFeeds	R	Open	None(0)
CapRecPageMode	bool	R	Open	TRUE
CapSlpPresent	bool	R	Open	FALSE
CapSlpFullslip	bool	R	Open	FALSE
CapSlp2Color	bool	R	Open	FALSE
CapSlpBarCode	bool	R	Open	FALSE
CapSlpBitmap	bool	R	Open	FALSE
CapSlpBold	bool	R	Open	FALSE
CapSlpDhigh	bool	R	Open	FALSE
CapSlpDwide	bool	R	Open	FALSE
CapSlpDwideDhigh	bool	R	Open	FALSE
CapSlpEmptySensor	bool	R	Open	FALSE
CapSlpItalic	bool	R	Open	FALSE
CapSlpLeft90	bool	R	Open	FALSE
CapSlpNearEndSensor	bool	R	Open	FALSE
CapSlpRight90	bool	R	Open	FALSE
CapSlpRotate180	bool	R	Open	FALSE
CapSlpUnderline	bool	R	Open	FALSE
CapSlpBothSidesPrint	bool	R	Open	FALSE
CapSlpCartridgeSensor	PrinterCartridgeSensors	R	Open	None(0)
CapSlpColor	PrinterColors	R	Open	None(0)
CapSlpPageMode	bool	R	Open	FALSE
CapSlpRuledLine	LineDirection	R	Open	None(0)
AsyncMode	bool	R/W	Open	FALSE
CartridgeNotify	PrinterCartridgeNotify	R/W	Open	Disabled (0) 書込み不可
CharacterSet	int	R/W	Open,Claim & Enable	初期値は PosExplorer 外部設定値によって変化 Enabled 後に書込み可能
CharacterSetList	int[]	R	Open	初期値は PosExplorer 外部設定値によって変化
CoverOpen	bool	R	Open,Claim & Enable	FALSE
ErrorLevel	PrinterErrorLevel	R	Open	None(1)
ErrorStation	PrinterStation	R	Open	None(0)
ErrorString	string	R	Open	“”
FontTypefaceList	string[]	R	Open	“Arial, Times New Roman”
FlagWhenIdle	bool	R/W	Open	FALSE
MapCharacterSet	bool	R/W	Open	TRUE
MapMode	MapMode	R/W	Open	Dots (1) Open 後に書込み可能

専用	型	ｱｸｾｽ	初期化	初期値と備考
PageModeArea	System.Drowin g. Point	R	Open	※初期値はPosExplorer外部設定値によって変化
PageModeDescriptor	PageModeDescriptors	R	Open	Bitmap + Barcode + BitmapRotate + BarcodeRotate
PageModeHorizontalPosition	int	R/W	Open	0
PageModePrintArea	System.Drowin g. Rectangle	R/W	Open	Rectangle(0,0)
PageModePrintDirection	PageModePrintDirection	R/W	Open	None(0)
PageModeStation	PrinterStation	R/W	Open	None(0)
PageModeVerticalPosition	int	R/W	Open	0
RotateSpecial	Rotation	R/W	Open	Normal (1) Open 後に書込み可能
JrnLineChars	int	R/W	Open,Claim & Enable	0 書込み不可
JrnLineCharsList	int[]	R	Open	
JrnLineHeight	int	R/W	Open,Claim & Enable	0 書込み不可
JrnLineSpacing	int	R/W	Open,Claim & Enable	0 書込み不可
JrnLineWidth	int	R	Open,Claim & Enable	0
JrnLetterQuality	bool	R/W	Open,Claim & Enable	FALSE 書込み不可
JrnEmpty	bool	R	Open,Claim & Enable	FALSE
JrnNearEnd	bool	R	Open,Claim & Enable	FALSE
JrnCartridgeState	PrinterCartridgeStates	R	Open,Claim & Enable	OK(0)

専用	型	ｱｸｾｽ	初期化	初期値と備考
JrnCurrentCartridge	PrinterColors	R/W	Open,Claim & Enable	None(0) 書込み不可
RecLineChars	int	R/W	Open,Claim & Enable	初期値は PosExplorer 外部設定値によって変化 Open 後に書込み可能
RecLineCharsList	int[]	R	Open	初期値は PosExplorer 外部設定値によって変化
RecLineHeight	int	R/W	Open,Claim & Enable	初期値は PosExplorer 外部設定値によって変化 書込み不可
RecLineSpacing	int	R/W	Open,Claim & Enable	初期値は PosExplorer 外部設定値によって変化 Open 後に書込み可能
RecLineWidth	int	R	Open,Claim & Enable	初期値は PosExplorer 外部設定値によって変化
RecLetterQuality	bool	R/W	Open,Claim & Enable	TRUE Open 後に書込み可能
RecEmpty	bool	R	Open,Claim & Enable	FALSE
RecNearEnd	bool	R	Open,Claim & Enable	FALSE
RecSidewaysMaxLines	int	R	Open,Claim & Enable	初期値は PosExplorer 外部設定値によって変化
RecSidewaysMaxChars	int	R	Open,Claim & Enable	初期値は PosExplorer 外部設定値によって変化
RecLinesToPaperCut	int	R	Open,Claim & Enable	初期値は PosExplorer 外部設定値によって変化
RecBarCodeRotationList	Rotation[]	R	Open	Normal (0), Right90(257), Left90(258), Rotate180(259)
RecCartridgeState	PrinterCartridgeStates	R	Open,Claim & Enable	Unknown(68435456)
RecCurrentCartridge	PrinterColors	R/W	Open,Claim & Enable	None(0) 書込み不可能

専用	型	ｱｸｾｽ	初期化	初期値と備考
RecBitmapRotationList	Rotation[]	R	Open	Normal (0), Right90(257), Left90(258), Rotate180(259)
SlpLineChars	int	R/W	Open,Claim & Enable	0 書込み不可能
SlpLineCharsList	int[]	R	Open	
SlpLineHeight	int	R/W	Open,Claim & Enable	0 書込み不可能
SlpLineSpacing	int	R/W	Open,Claim & Enable	0 書込み不可能
SlpLineWidth	int	R	Open,Claim & Enable	0
SlpLetterQuality	bool	R/W	Open,Claim & Enable	FALSE 書込み不可能
SlpEmpty	bool	R	Open,Claim & Enable	FALSE
SlpNearEnd	bool	R	Open,Claim & Enable	FALSE
SlpSidewaysMaxLines	int	R	Open,Claim & Enable	0
SlpSidewaysMaxChars	int	R	Open,Claim & Enable	0
SlpMaxLines	int	R	Open,Claim & Enable	0
SlpLinesNearEndToEnd	int	R	Open,Claim & Enable	0
SlpBarCodeRotationList	Rotation[]	R	Open	
SlpPrintSide	int	R	Open,Claim & Enable	0
SlpCartridgeState	int	R	Open,Claim & Enable	0

専用	型	ｱｸｾｽ	初期化	初期値と備考
SlpCurrentCartridge	int	R/W	Open,Claim & Enable	0 書込み不可能
SlpBitmapRotationList	Rotation[]	R	Open	

※アクセスの R は読込み専用、R/W は読込み、書き込み可能であることを示します。また、初期化は、初期化に必要なメソッド・プロパティを示し、Open は Open メソッド、Claim は Claim メソッド、Enable は DeviceEnabled プロパティを TRUE に設定することを意味します。また、条件が Open&Claim および Open,Claim&Enable 状態になっているプロパティについては、Open メソッド実行後に取得可能となりますが、Open,Claim &Enable 後にはじめて値が初期化される場合があるため、プロパティ取得時には条件を満たした上でアクセスを行なってください。

メソッド

共通	必要条件
Open	--
Close	Open
Claim	Open
Release	Open, Claim
ClearOutput	Open, Claim & Enable
CheckHealth	Open, Claim & Enable
DirectIO	Open, Claim & Enable
ResetStatistics	Open, Claim & Enable
RetrieveStatistics	Open, Claim & Enable
UpdateStatistics	Open, Claim & Enable

専用	必要条件
PrintNormal	Open, Claim & Enable
PrintTwoNormal	Open, Claim & Enable
PrintImmediate	Open, Claim & Enable
BeginInsertion	Open, Claim & Enable
EndInsertion	Open, Claim & Enable
BeginRemoval	Open, Claim & Enable
EndRemoval	Open, Claim & Enable
CutPaper	Open, Claim & Enable
RotatePrint	Open, Claim & Enable
PrintBarCode	Open, Claim & Enable
PrintBitmap	Open, Claim & Enable
TransactionPrint	Open, Claim & Enable
ValidateData	Open, Claim & Enable
SetBitmap	Open, Claim & Enable
SetLogo	Open, Claim & Enable
ChangePrintSide	Open, Claim & Enable
MarkFeed	Open, Claim & Enable
ClearPrintArea	Open, Claim & Enable
PageModePrint	Open, Claim & Enable
PrintMemoryBitmap	Open, Claim & Enable

イベント

名称	必要条件
DirectIOEvent	Open, Claim & Enable
ErrorEvent	Open, Claim & Enable
OutputCompleteEvent	Open, Claim & Enable
StatusUpdateEvent	Open, Claim & Enable

4．2．印字データとエスケープシーケンス

本クラスライブラリは以下のエスケープシーケンスをサポートします。

1）指定時のみ、動作するエスケープシーケンス

名称	データ	内容
用紙カット	ESC |#P	レシート用紙を切ります。文字'#'は、要求されたカットのパーセンテージを示す ASCII10 進数の文字列です。'#'を省略することも可能です。'1'～'99'の間はパーシャルカットが、'100'あるいは省略されている場合にはフルカットとなります。'1'～'100'以外の場合は無視されます。尚、POS プリンターにバッファリングされているデータがある場合（印字要求をしたが、POS プリンターに印字されていない場合）は、カットすることができません。レシートカットを行うためには行の先頭である必要があります。 また、**RotatePrint** メソッドによる左右 90 度回転中では無効となり左右回転解除後に動作します。
フィードと用紙カット	ESC |#fP	**RecLinesToPaperCut** 行分の紙送りをした後、レシート用紙をカットします。文字'#'は、「用紙カット」エスケープシーケンスで定義されています。尚、POS プリンターにバッファリングされているデータがある場合（印字要求をしたが、POS プリンターに印字されていない場合）は、カットすることができません。レシートカットを行うためには行の先頭である必要があります。 また、**RotatePrint** メソッドによる左右 90 度回転中では無効となり左右回転解除後に動作します。
フィードとカットとスタンプ印刷	ESC |#sP	サポートしません。
ビットマップ印刷	ESC |#B	**SetBitmap** メソッドで保存したビットマップを印刷します。'#'はビットマップ番号で、'1'-,'20'までの 20 個のビットマップ印刷をサポートしています。印字時に **RecLetterQuality** プロパティの値を変更することで印字品質の変更が可能です。印字品質の取扱については、**PrintBitmap** メソッドと同様です。'#'を省略した場合は、文字列として扱われます。 **RotatePrint** メソッドによる左右 90 度回転中も印字動作は行いますが、ビットマップサイズによる印字領域の計算は行われないため、正常に印字されない場合が有ります。 "#"が省略された場合は、"B"から始まる文字列データとみなします。 "#"に**SetBitmap**メソッドで保存していない番号を指定した場合、プリンターに対してはプリントコマンドを発行しますが、印字はされません。
トップロゴ印刷	ESC |tL	**SetLogo** メソッドで保存したトップロゴを印刷します。
ボトムロゴ印刷	ESC |bL	**SetLogo** メソッドで保存したボトムロゴを印刷します。
スタンプ印刷	ESC |sL	サポートしません。

名称	データ	内容
複数行フィード	ESC |#lF	複数行のフィードをします。文字'#'は、フィードする行数を示す ASCII10 進数の文字列です。もし、'#'が省略されていれば、1 行分のフィードをします。'#'は'1'-''255'までの値をサポートします。印字データがなければ改行量にしたがった改行動作を実行し、印字データがある場合には、印字データの高さ分の改行動作を実行します。'#'の設定値が 35.4 インチ(約 900mm)を超える場合は、35.4 インチ(約 900mm)の紙送りを実行します。 **RotatePrint** メソッドによる左右 90 度回転中は、次回印字位置がフィード指定した行数を改行した後に印字されます。
単位フィード	ESC |#uF	**MapMode** で指定された単位でフィードをします。文字'#'は、フィード量を示す ASCII10 進数の文字列です。もし、'#'が省略されていれば、各 **MapMode** における 1 単位分のフィードをします。 **MapMode** が Dots(1)の場合、'#'は'1'-''127'までの値をサポートします。('1'より小さい値の場合はコマンドを実行せず、`127`より大きい値は、`127`が指定されたものとして実行されます) **MapMode** が Twips (2)の場合、'#'は'1'-'903'までの値をサポートします。('1'より小さい値の場合はコマンドを実行せず、'903'より大きい値は、'903'が指定されたものとして実行されます) **MapMode** が English (3)の場合、'#'は'1'-''627'までの値をサポートします。('1'より小さい値の場合はコマンドを実行せず、' 627'より大きい値は、'627'が指定されたものとして実行されます) **MapMode** が Metric (4)の場合、'#'は'1'-''1594'までの値をサポートします。('1'より小さい値の場合はコマンドを実行せず、'1594'より大きい値は、'1594'が指定されたものとして実行されます) 改行量は、プリンターに設定されている改行量に影響されません。また、行の途中で実行され、指定フィード数が 1 改行未満の場合は、1 行改行されます。 **RotatePrint** メソッドによる左右 90 度回転中は、次回印字位置が単位フィード指定した値分だけ、間が開いた後に印字されます。
逆フィード	ESC |#rF	サポートしません。

名称	データ	内容
埋め込みデータの送信	ESC |#E	"#E"に続く文字列は、一切変更されずにデバイスへ渡されます。'#'は、ASCII10 進数の文字列で、このエスケープシーケンスに続く、そのままデバイスへ渡されるべき文字列のバイト数を指定します。バイト数の上限は int の値の整数値(2147483647)です。'#'が省略された場合は、エスケープシーケンスとはみなされず、印字データとしてみなされます。文字列が制御コード及び 80H～FFH のコードは、期待通りに出力することが出来ません。この場合は、NIBBLE 形式で印字データを設定してください。 '#'で指定された分の印字データがエスケープシーケンスの指定後に設定されていない場合は、送信可能な印字データのみを送信します。（例：ESC|2E"a" が指定された場合、1 バイトしか文字列が設定されてないため、"a"のみを送信します。） また、**RotatePrint** メソッドによる左右 90 度回転中は、埋め込みデータの送信で指定されたデータ列は文字列としてカウントされないため、横幅が正確に計算できなくなります。この場合は空白等の挿入で印字幅の調整を行なってください。
バーコード印刷 （次頁を参照）	ESC|#R	バーコードを印刷します。文字'#'は、ASCII10 進数の文字列で、R に続く文字列（バーコードの特性を定義）の文字数を指定します。詳細は欄外を参照してください。 **RotatePrint** メソッドによる左右 90 度回転中に印字される場合がありますが、指定バーコード幅による印字領域の計算は行われないため、正常に印字されない場合が有ります。印字される場合は、その他指定されている文字列データがバーコード幅を超えた場合です。 **RotatePrint** メソッドで BarCode 指定した場合は、正常にバーコードを印字しません。 **RotateSpecial** プロパティでは回転印刷を行いません。 パラメータで指定可能な幅は、**RecLineWidth** プロパティの値までとなり、**RotateSpecial** プロパティの影響は受けません。

アプリケーションは ESC|#R エスケープシーケンスを利用することにより、バーコードを印刷することができます。文字'#'は、R に続く文字列（バーコードの特性を定義）の文字数を指定します。
R に続く文字列では、小文字のアルファベットと数値を使用して、バーコードの特性を指定します。数値には、PrintBarCode メソッドのために定義された定数値が利用できます。
属性を示す文字は以下の通りです。

s	symbology（バーコードタイプ）
h	height（バーコードの高さ）
w	width（バーコードの幅）
a	alignment（バーコードの位置）
t	text position（HRI 文字列の位置）
d	start of data（バーコードデータの開始位置）
e	end of data（バーコードデータの終了位置）

属性は、上記のリストで示された順番で必ず記述しなければなりません（省略不可）。UPC-A を、センタリング、バーコードの下に HRI 文字列を印字、200dot 高さ、400dot 幅、の条件で印刷する場合の例を以下に示します。
ESC|33Rs101h200w400a-2t-13d123456789012e

また、各パラメータの閾値は下記のようになります。閾値をオーバーした場合には、バーコードは印字されません。

バーコード	Width(dot)	Height(dot)	Alignment
2 次元バーコード以外	個々のバーコードの最小幅 ～ **RecLineWidth**	1 ～ 255	**PrintBarcode** メソッドで定義されているすべての値が指定可能
PDF417	172 ～ **RecLineWidth**	12 ～ 831	**PrintBarcode** メソッドで定義されているすべての値が指定可能
PDF417 **RotatePrint** メソッド 左右 90 度回転中	172 ～ 831	12 ～ **RecLineWidth**	指定した値に限らず、すべて左寄せとなる
QR	21 ～ **RecLineWidth**	1 ～ 16 (モジュール幅として)	**PrintBarcode** メソッドで定義されているすべての値が指定可能
QR **RotatePrint** メソッド 左右 90 度回転中	21 ～ **RecLineWidth**	1 ～ 16 (モジュール幅として)	指定した値に限らず、すべて左寄せとなる
MicroQR	11 ～ RecLineWidth	1 ～ 16 (モジュール幅として)	PrintBarcode メソッドで定義されているすべての値が指定可能
MicroQR RotatePrint メソッド 左右 90 度回転中	11 ～ RecLineWidth	1 ～ 16 (モジュール幅として)	指定した値に限らず、すべて左寄せとなる

2）印字中に、動作するエスケープシーケンス

明確に変更されるまで、その状態を保つ属性があります。

.NET版では該当のエスケープシーケンスはありません。

3）印字時に、動作するエスケープシーケンス

各印刷メソッドの終わりや、あるいは”ノーマル”シーケンスによってリセットされる属性があります。

名称	データ	内容
ボールド	ESC |(!)bC	ボールド体で印刷します。 '!'が指定された場合は、ボールドを無効にします。
アンダーライン	ESC |#uC	アンダーラインと共に印刷します。文字'#'は、アンダーラインの太さ(ドット単位)を示す ASCII10 進数の文字列です。 1 ドット、2 ドットのアンダーラインのみサポートしています。省略時は、1 ドットのアンダーラインを印刷します。 フォントタイプ指定で 2 以上を指定した場合は、2 ドットを指定しても、1 ドットのアンダーラインを印字します。
イタリック	ESC |(!)iC	イタリック体で印刷します。 フォントタイプ指定で 2 以上を指定した場合のみサポートします。 '!'が指定された場合は、イタリックを無効にします。。
カスタムカラー	ESC |#rC	サポートしません。
赤色	ESC |rC	レシートの第 2 色目で印字します。 プリンター設定の”印字色”が”2 色”に設定されているときのみ印字が可能です。 設定プログラムにて、Color = mono に指定されている場合は、このエスケープシーケンス指定による影響はありません。
反転文字	ESC |(!)rvC	明暗を反対にして印刷します。 '!'が指定された場合は、反転文字を無効にします。
網掛け文字	ESC |#sC	サポートしません。
取消線	ESC |(!)stC	取消線を付加して印刷します。 '!'が指定された場合は、取消線を無効にします。 フォントタイプ指定で 0 または 1 を指定した場合は、2 重取消線で印刷されます。
縦横 1 倍角	ESC |1C	通常の大きさで印刷します。
横倍角	ESC |2C	横倍角文字で印刷します。
縦倍角	ESC |3C	縦倍角文字で印刷します。
縦横倍角	ESC |4C	縦横倍角文字で印刷します。
横倍率	ESC |#hC	文字を横方向に拡大して印刷します。文字”#”は、横方向の拡大倍率を示す ASCII10 進数の文字列で、1～8 倍までサポートしています。 “#”省略時は 1 倍の大きさで印刷します。
縦倍率	ESC |#vC	文字を縦方向に拡大して印刷します。文字”#”は、縦方向の拡大倍率を示す ASCII10 進数の文字列で、1～8 倍までサポートしています。 “#”省略時は 1 倍の大きさで印刷します。

名称	データ	内容
色指定	ESC |#fC	'#'カラーで印刷します。代替文字'#'は、RGB を示す ASCII10 進数で置きかえられます。それらは、赤、緑、青を示す 3 桁です。例えば、 "255255000"は、黄色になります。 RGB のそれぞれの 3 桁の数値は 0 から 255 までを指定することができます。もし、'#'が省略されていれば、黒色で印字されます。 ビットマップ印刷には影響しません。指定した値が、RGB で許容された範囲を超える場合、色指定のエスケープシーケンスは無視されます。 PosExplorer 外部設定値"RasterMode"="F"
中央揃え	ESC |cA	中心に以下のテキストを整列させます。行の先頭で指定しなければ有効となりません。 RotatePrint メソッドによる左右 90 度回転中は無効となります。 PageModePrint メソッドによるページモード開始中は無効となります。
右寄せ	ESC |rA	右に以下のテキストを整列させます。行の先頭で指定しなければ有効となりません。 RotatePrint メソッドによる左右 90 度回転中は無効となります。 PageModePrint メソッドによるページモード開始中は無効となります。
ノーマル	ESC |N	POS プリンターの属性を通常の状態に回復させます。行の先頭でなければ、中央揃え、右寄せを回復させることはできません。
サブスクリプト	ESC |(!)tbC	サポートしません。
スーパースクリプト	ESC |(!)tpC	サポートしません。

4．3．共通プロパティ

POS プリンターに対して共通に提供されるプロパティを説明します。
プロパティは読込み専用のものと、読込み、書き込み可能なものの 2 種類が存在します。書き込み可能なプロパティには、R/W をプロパティ名の横に記述します。
また、PosControlException により通知される例外は特別な意味合いがあるものについてのみ明記します。

Compatibility プロパティ、SynchronizingObject プロパティについては MSDN を参照ください。

“https://docs.microsoft.com/en-us/previous-versions/windows/embedded/ms845351(v=winembedded.10)”
“https://docs.microsoft.com/en-us/previous-versions/windows/embedded/ms845348(v=winembedded.10)”

CapCompareFirmwareVersion プロパティ

形式

bool CapCompareFirmwareVersion;

説明

TRUE ならばサービス/デバイスは、ファームウェアファイルのバージョンと物理デバイス内のファームウェアバージョンを比較する機能を有します。
このプロパティは **Open** メソッドにより **TRUE** に初期化されます。

CapPowerReporting プロパティ

形式

PowerReporting **CapPowerReporting;**

説明

デバイスの電源通知能力を識別します。電源通知能力を示す値は下記の通りです。

値	意味
Standard (1)	SO は 2 種類の電源状態を判断し、通知が可能です。(ONLINE と OFF_OFFLINE)

このプロパティは **Open** メソッドにより初期化されます。

CapStatisticsReporting プロパティ

形式

bool CapStatisticsReporting;

説明

このプロパティは **Open** メソッドにより **FALSE** に初期化されます。スタティスティクスの通知はサポートしません。

CapUpdateStatistics プロパティ

形式

bool CapUpdateStatistic;

説明

このプロパティは **Open** メソッドにより **FALSE** に初期化されます。スタティスティクスの通知はサポートしません。

CheckHealthText プロパティ

形式

string CheckHealthText;

説明

直前に呼び出した **CheckHealth** メソッドの結果を保持します。以下に診断結果の例を示します

- Internal の場合　“Internal HCheck: Successful”　“Internal HCheck: OFF/OFFLINE”
- External の場合　“External HCheck：Not Supported”
- Interactive の場合　“Interactive HCheck：Not Supported“

最初の **CheckHealth** メソッド呼び出し以前に、この値は（空文字）に初期化されています。

Claimed プロパティ

形式

bool Claimed;

説明

TRUE：デバイスの排他アクセス権が獲得されています。

FALSE：デバイスはほかのアプリケーションと共有できるよう解放されています。

Claimed プロパティの値は **Open** メソッドによって **FALSE** に初期化されます。

DeviceDescription プロパティ

形式

string DeviceDescription;

説明

“FP 1 Station Thermal POSPrinter (C)20xx- Fujitsu Isotec” が設定されています。

このプロパティはデバイスを識別する文字列で、デバイスとそれに関連する情報を示します。

このプロパティは **Open** メソッドによって初期化されます。

DeviceEnabled プロパティ R/W

形式

bool DeviceEnabled;

説明

TRUE :

デバイスはイネーブルにされています（動作状態です）。**TRUE** に変更するとイネーブルにされます。

FALSE :

デバイスはディセーブルにされています。**FALSE** に変更すると、ディセーブルにされます。

デバイスを使用する前に、アプリケーションはこのプロパティを **TRUE** に設定しなければなりません。
また、**DeviceEnabled** が **TRUE** の間、デバイスの接続状態(**PowerReporting**)が通知されます。このプロパティは **Open** メソッドにより **FALSE** に初期化されます。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
NotClaimed (103)	排他使用デバイスをイネーブルにするには、その前に排他アクセス権を獲得しなければなりません。
NoHardware (107)	POS プリンターがオフ／オフライン、またはケーブルが接続されていません。これらを解消した後、もう一度実行してください。（※USB インターフェースの場合、POS プリンターを接続していても、POS プリンターに設定されているシリアル番号とレジストリに設定されたシリアル番号が異なる場合は、この例外が発生します。） ※LAN インターフェースの場合、ネットワーク上に POS プリンターを接続していても、POS プリンターの IP アドレスとレジストリに設定された IP アドレスが異なる場合は、この例外が発生します。
Failure (111)	接続ポートのオープンに失敗しました。他のプログラムから接続ポートが使用されていないか、接続ポートが存在しているかを確認した後、もう一度実行してください。
Timeout (112)	POS プリンターに接続できませんでした。カバーが開いているか、用紙切れの可能性があります。
Busy (113)	処理中の為、プロパティの設定に失敗しました。処理終了後にプロパティを設定してください。
その他	**ErrorCode** プロパティを参照してください。

DeviceName プロパティ

形式

string DeviceName;

説明

"FP 1 Station Thermal POSPrinter" が設定されています。

このプロパティは、デバイスとそれに関連する情報を示します。**DeviceDescription** プロパティの短縮バージョンで、30 文字までに制限されています。このプロパティは **Open** メソッドにより初期化されます。

FreezeEvents プロパティ R/W

形式

bool FreezeEvents;

説明

TRUE の場合、コントロールからイベントが通知されません。

凍結が解除されるまで、イベントはコントロールが保持しています。

FALSE の場合、コントロールからイベントが通知されます。凍結されていた間に保持されたイベントがあれば、**FreezeEvents** を **FALSE** に変更すると、そのイベントが通知されます。

イベントによる割り込みが望ましくない状況で、アプリケーションはイベントの凍結を選択できます。**ErrorEvent** を凍結した場合、**State** プロパティが Busy**(3)**となります。この場合、コントロールをクローズすることができなくなりますので、この場合は、**ClearOutput** メソッドで凍結中のイベントを破棄するか、**TRUE** に設定して、**ErrorEvent** を発生させた後、**Close** メソッドを実行してください。

このプロパティは **Open** メソッドにより **FALSE** に初期化されます。

OutputId プロパティ

形式

int OutputId;

説明

非同期要求（**AsyncMode** プロパティが **TRUE** に設定されているときの、非同期対応メソッドの呼び出し）を一意に識別するための識別子を保持しています。

メソッドが同期・非同期出力の開始に成功すると、コントロールはリクエストに識別子を割り当てます。非同期出力の場合、出力が完了すると、**OutputCompleteEvent** を通知して、その出力 ID をパラメータとして渡します。

出力 ID 番号は、1 ～65535 の間でサイクリックに採番されます。

PowerNotify プロパティ R/W

形式

PowerNotification PowerNotify;

説明

アプリケーションから設定された電源通知機能のタイプです。

電源通知機能を示す値は下記のとおりです。

値	意味
Disabled(0)	コントロールはアプリケーションに対していかなる電源通知を提供しません。電源通知に関する **StatusUpdateEvent** は通知されず、また **PowerState** プロパティには何も設定されません。 （デフォルト値）
Enabled(1)	**DeviceEnabled** が **TRUE** に設定されると、コントロールは電源通知に関する **StatusUpdateEvent** の通知、**PowerState** プロパティの更新が行われます。

PowerNotify プロパティは、デバイスがディセーブルの間、すなわち **DeviceEnabled** プロパティが **FALSE** の間のみ設定が可能です。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
Illegal(106)	次のいずれかが発生しました： ・ デバイスは既にイネーブル。 ・ プロパティ設定値が不正。
その他	エラーコード一覧を参照してください。

PowerState プロパティ

形式

PowerState PowerState;

説明

PowerNotify が Enabled**(1)**の間、現在のデバイスの電源状態が設定されます。電源状態を示す値は下記のとおりです。

値	意味
Unknown(2000)	下記の理由のいずれかにより、デバイスの電源状態の判断ができません。（デフォルト値） **PowerNotify** = **Disabled(0)**で電源通知機能がディセーブル。 **DeviceEnabled** = **FALSE**、電源状態監視はデバイスがイネーブルになるまで動作せず。
Online(2001)	デバイスは電源オンでかつレディ状態です。
OffOffline(2004)	デバイスは電源オフ、または本体に接続されていません。復旧方法については、**ErrorCode** の Nohardware(107)を参照してください。

このプロパティは、**Open** メソッドにより Unknown(2000)に初期化されます。**PowerNotify** が Enabled(1)にセットされ、**DeviceEnabled** が TRUE の場合、このプロパティは SO が検出した電源状態の検出に従い更新されます。

ServiceObjectDescription プロパティ

形式

string ServiceObjectDescription;

説明

“Fujitsu Isotec FP POS Printer Service Object”が設定されています。
このプロパティは **Open** メソッドによって初期化されます。

ServiceObjectVersion プロパティ

形式

System.Version ServiceObjectVersion;

説明

“1013XXX” が設定されています。サービスオブジェクトバージョン番号を示します。（XXX は、サービスオブジェクトの提供時期によって異なります。）このプロパティは **Open** メソッドによって初期化されます。

State プロパティ

形式

ControlState State;

説明

コントロールの現在の状態を示します。

値	意味
Closed(1)	コントロールはクローズしています。（デフォルト）
Idle(2)	コントロールは正常な状態にあり、ビジーではありません。
Busy(3)	コントロールは正常な状態にあり、出力を実行しているためビジーです。
Error(4)	エラーが報告され、通常の I/O を再開するには、その前にアプリケーションがコントロールを正常な状態に戻さなければなりません。

このプロパティはいつでも読取り可能です。

4．4．共通メソッド

POS プリンターに対して共通に提供されるメソッドを説明します。

CheckHealth メソッド

形式

string CheckHealth (HealthCheckLevel *Level*);

Level パラメータは、デバイスで実行するヘルスチェックのタイプを示します。以下の値を指定できます。

値	意味
Internal(1)	オンラインチェックを行います。結果を以下のように **CheckHealthText** プロパティに設定します。 POS プリンターが POS に接続されている且つ、電源 ON の場合、**CheckHelth** プロパティに“Internal HCheck: Successful”が設定されます。 POS プリンターが POS に接続されていない、あるいは電源が入っていない場合、**CheckHelth** プロパティに“Internal HCheck: OFF/OFFLINE”が設定されます。
External(2)	本クラスライブラリではサポートされません。 **CheckHelth** プロパティに" External Hcheck:Not Supported "が設定されます。 Illegal(106)を返します。
Interactive(3)	本クラスライブラリではサポートされません。 **CheckHelth** プロパティに" External Hcheck:Not Supported "が設定されます。 Illegal(106)を返します。

説明

デバイスの状態をテストするときに呼び出します。このメソッドの結果は **CheckHealthText** プロパティに格納されます。

CheckHealth メソッドは常に同期です。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
例外なし	ヘルスチェックプロシージャが適切に開始されたことを示し、確認できた場合、デバイスが正常であることを示します。しかし、正常かどうかはテスト結果を見ないと決定できません。
Illegal(106)	サポートされていない *Level* パラメータが指定されました。
NoHardware(107)	Internal(1)を指定して実行した結果、オフラインでした。
Busy(113)	出力が進行中の間、実行できません。
その他	**ErrorCode 一覧を参照してください。**

Claim メソッド

形式

void Claim (int *Timeout*);

Timeout パラメータは、排他アクセス権を獲得するまでの最大待ち時間（ミリ秒単位）を示します。

ゼロの場合、メソッドはデバイスの排他アクセスが獲得できなかった場合でも、その結果を直ちに戻します。

OPOS_FOREVER(-1)が設定されている場合は、メソッドは排他アクセス権が獲得できるまで必要なだけ待ちます。

説明

デバイスに対して排他アクセスを要求するときに、このメソッドを呼び出します。

POS プリンターデバイスは排他アクセス権を獲得しなければ、使用することはできません。

成功すると、**Claimed** プロパティは **TRUE** に設定されます。

Claim メソッドを実行すると、POS プリンターデバイスとの接続を確立し、処理可能な状況であることを確認します。処理可能であれば、固定データを要求し、**Claim** メソッドは正常終了します。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
Illegal(106)	無効な ***Timeout*** パラメータが指定されています。
Timeout(112)	別のアプリケーションがデバイスに排他アクセスしていて、解放されるのを待っていましたが ***Timeout*** 時間（ミリ秒単位）が過ぎました。もしくは、***Timeout*** 時間（ミリ秒）経過しても、POS プリンターデバイスが処理可能な状態になりませんでした。

ClearOutput メソッド

形式

void ClearOutput ();

説明

PrintNormal, **CutPaper**, **RotatePrint**, **PrintBarCode**, **PrintBitmap**, **TransactionPrint**, **PageModePrint** の各メソッドの非同期発行によりバッファリングされている全てのデバイス出力をすべてクリアするときに呼び出します。また、**RotatePrint** メソッド、**TransactionPrint** メソッド、**PageModePrint** メソッドにより回転モード、一括処理モード、ページモード中の場合、解除を行ないます。

保留になっていた出力エラーイベント（**FreezeEvents** が **FALSE** に設定されるのを待ち合わせているもの）もクリアされます。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
例外なし	出力がクリアされました。
Failure(111)	デバイスは他のプロセスにより排他アクセスされています。
その他	**ErrorCode 一覧を参照してください。**

Close メソッド

形式

void Close ();

説明

デバイスとそのリソースを解放するときに呼び出します。

DeviceEnabled プロパティが **TRUE** の場合、まずデバイスがディセーブルにされます。

Claimed プロパティが **TRUE** の場合、まずデバイスの排他アクセスが解除されます。

イベント処理中（イベントハンドラ内）では実行しないでください。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
例外なし	デバイスがディセーブルにされ、クローズしました。
Busy(113)	非同期処理が実行中です。
その他	**ErrorCode 一覧を参照してください。**

DirectIO メソッド

形式

DirectIOData DirectIO (int *command*, int *data*, object *obj*);

説明

本クラスライブラリでは、Binary 文字列の同期・非同期送信, 16 進数文字列の変換同期・非同期送信をサポートします。

Binary 文字列の同期送信

command	OPOS_FIT_DIO_BIN_SYNC(0) OPOS_FIT_DIO_BIN_IMMIDIATE(1) OPOS_FIT_DIO_BIN_REALTIME(2)
data	未使用
obj	IN Binary 文字列
機能	Binary 文字列を POS プリンターに同期送信します。このコマンドにより、POS プリンターに直接コマンドを送信することが可能となります。 Binary 文字列は以下のフォーマットとなります。 &H00 ～ &Hff(&HFF) プリンター側にパーシャルカットコマンド(1BH 6DH)を送信する場合には以下のように設定を行います。 “&H1B&H6D”(アルファベットは小文字も許容) HEX として認められない文字は無視します。 例： “ABCD&H00EFG” を送信した場合、プリンター側には、”0x00” のみが送信されます。

16 進数文字列の変換同期送信

Command	OPOS_FIT_DIO_HEX_SYNC(5)
pData	未使用
pString	IN　16 進数文字列
機能	16 進数文字列をプリンターに同期送信します。このコマンドにより、プリンターに直接コマンドを送信することが可能となります。 １バイトを 2 バイトのキャラクタ文字で表します。 &H00 → ‘00’ キャラクタは、’0’～’9’,‘A’～’F’, ‘a～’f’が有効です。 上記以外のキャラクタが 1 つでも存在した場合は、コマンドを送信せず、Illegal(106)を返します。

Binary 文字列の非同期送信

Command	OPOS_FIT_DIO_BIN_ASYNC(6)
pData	未使用
pString	IN　Binary 文字列
機能	Binary 文字列をプリンターに非同期送信します。このコマンドにより、プリンターに直接コマンドを送信することが可能となります。実行後、**OutputCompleteEvent** が上がります。 Binary 文字列は以下のフォーマットとなります。 &H00 ～ &Hff(&HFF) プリンター側にパーシャルカットコマンド(1BH 6DH)を送信する場合には以下のように設定を行います。 “&H1B&H6D”(アルファベットは小文字も許容) HEX として認められない文字は無視します。 例： “ABCD&H00EFG” を送信した場合、プリンター側には、”0x00” のみが送信されます。

16 進数文字列の変換非同期送信

Command	OPOS_FIT_DIO_HEX_ASYNC(7)
pData	未使用
pString	IN　16 進数文字列
機能	16 進数文字列をプリンターに非同期送信します。このコマンドにより、プリンターに直接コマンドを送信することが可能となります。実行後、OutputCompleteEvent が上がります。 １バイトを 2 バイトのキャラクタ文字で表します。 &H00 → ‘00’ キャラクタは、’0’～’9’,‘A’～’F’, ‘a～’f’が有効です。 上記以外のキャラクタが 1 つでも存在した場合は、コマンドを送信せず、Illegal(106)を返します。

QR コードエラー訂正レベル設定

Command	OPOS_FIT_DIO_SET_QRERRORLV(20)
pData	IN　QR コードエラー訂正レベル
pString	未使用
機能	QR コードのエラー訂正レベルを設定します。 デフォルト値は 0 です。DeviceEnabled 実行時にデフォルトに設定されます。

値	復元能力
0	レベル L(7%)
1	レベル M(15%)
2	レベル Q(25%)
3	レベル H(30%)

QR コードエラー訂正レベル取得

Command	OPOS_FIT_DIO_GET_QRERRORLV(21)
pData	OUT　QR コードエラー訂正レベル
pString	未使用
機能	QR コードのエラー訂正レベルを取得します。実行後、obj に QR コードのエラー訂正レベル(0～3)が設定されます。

マイクロ QR コードエラー訂正レベル設定

Command	OPOS_FIT_DIO_SET_MICROQRERRORLV(22)
pData	IN　マイクロ QR コードエラー訂正レベル
pString	未使用
機能	マイクロ QR コードのエラー訂正レベルを設定します。 デフォルト値は 0 です。DeviceEnabled 実行時にデフォルトに設定されます。

値	復元能力
0	レベル L(7%)
1	レベル M(15%)
2	レベル Q(25%)

マイクロ QR コードエラー訂正レベル取得

Command	OPOS_FIT_DIO_GET_MICROQRERRORLV(23)
pData	OUT　マイクロ QR コードエラー訂正レベル
pString	未使用
機能	マイクロ QR コードのエラー訂正レベルを取得します。実行後、obj にマイクロ QR コードのエラー訂正レベル(0～2)が設定されます。

PDF417 エラー訂正レベル設定

Command	OPOS_FIT_DIO_SET_PDFERROLV(24)
pData	IN　PDF417 エラー訂正レベル設定
pString	未使用
機能	PDF417 のエラー訂正レベルを設定します。 デフォルト値は 1 です。DeviceEnabled 実行時にデフォルトに設定されます。

値	詳細
1～40	データコードワード数 × 設定値 × 0.1 の四捨五入結果によって以下のエラー訂正レベルを設定します。 0～3：エラー訂正レベル 1 4～10：エラー訂正レベル 2 11～20：エラー訂正レベル 3 21～45：エラー訂正レベル 4 46～100：エラー訂正レベル 5 101～200：エラー訂正レベル 6 201～400：エラー訂正レベル 7 401 以上：エラー訂正レベル 8
48～56	エラー訂正レベル 0(48)　～ エラー訂正レベル 8(56)

PDF417 エラー訂正レベル取得

Command	OPOS_FIT_DIO_GET_PDFERROLV(25)
pData	OUT　PDF417 エラー訂正レベル設定
pString	未使用
機能	PDF417 コードのエラー訂正レベルを取得します。実行後、pData に現在設定されているエラー訂正レベルが設定されます。

PDF417 縦横サイズ設定

Command	OPOS_FIT_DIO_SET_PDFVHSIZE(26)
pData	IN　PDF417 縦横サイズ
pString	未使用
機能	PDF417 の縦横サイズを設定します。 デフォルト値は 0 です。DeviceEnabled 実行時にデフォルトに設定されます。 上位 16 ビット、下位 16 ビットにそれぞれ異なる値を設定します。 上位 16 ビット 　段数指定 0, 3 ～ 90 下位 16 ビット 　コードワード桁数指定 0～30 ※段数指定、コードワードが 0 の場合、PrintBarcode メソッドで指定される高さ、幅の制約条件において印字可能な値を自動で選択します。 ※段数×コードワード桁数が 928 を超える場合、バーコードの印刷はされません。

PDF417 縦横サイズ取得

Command	OPOS_FIT_DIO_GET_PDFVHSIZE(27)
pData	OUT　PDF417 縦横サイズ
pString	未使用
機能	PDF417 コードの縦横サイズを取得します。実行後、pData に現在設定されている縦横サイズが設定されます。

PDF417　1 段サイズ設定

Command	OPOS_FIT_DIO_SET_PDFSIZE (28)
pData	IN　PDF417　1 段サイズ
pString	未使用
機能	PDF417 の 1 段サイズを 1～8 の範囲で指定します。 実際は設定値にモジュール幅をかけた値になります。

PDF417　1 段サイズ取得

Command	OPOS_FIT_DIO_GET_PDFSIZE(29)
pData	OUT　PDF417　1 段サイズ
pString	未使用
機能	PDF417 コードの 1 段サイズを取得します。実行後、pData に現在設定されている 1 段サイズが設定されます。

PDF417　標準/簡易モード設定

Command	OPOS_FIT_DIO_SET_PDFMODE(30)
pData	IN　PDF417 標準/簡易モード設定
pString	未使用
機能	PDF417 のモードを標準/簡易モードに切り替えます。 0:標準モード 1:簡易モード

PDF417　標準/簡易モード取得

Command	OPOS_FIT_DIO_GET_PDFMODE(31)
pData	OUT　PDF417 標準/簡易モードサイズ
pString	未使用
機能	PDF417 コードのモードを取得します。実行後、pData に現在設定されているモードが設定されます。

Enable 後の全ての呼び出しに対して、これらコマンド以外の値が、***Command*** に設定された場合、Illegal(106)を返します。また、**DirectIO** メソッドは、**TransactionPrint**, **RotatePrint**, **PageModePrint** メソッドにてバッファリングされます。この場合、**DirectIO** メソッドの同期、非同期送信については、**TransactionPrint** メソッド、**RotatePrint** メソッド、**PageModePrint** メソッドが同期・非同期で実行されるかによって同期・非同期実行されます。

値	**意味**
Illegal(106)	このメソッドは有効ではありません
NoHardware(107)	POS プリンターはオフ、またはオフラインです。
Failure(111)	本クラスライブラリはエラー状態です。エラー状態を解除して実行してください。
Busy(113)	出力中なので実行できません。
Extended(114)	**ErrorCodeExtended** = CoverOpen(201): POS プリンターカバーが開いています。（同期メソッド実行時のみ） **ErrorCodeExtended** = RecEmpty(203):レシート用紙がありません。（同期メソッド実行時のみ） **ErrorCodeExtended** = Blackmark(10001):ブラックマークエラーが発生しました。（同期メソッド実行時のみ） **ErrorCodeExtended** = Fatal (10003):フェイタルエラーが発生しました。（同期メソッド実行時のみ） **ErrorCodeExtended** = Overheat (10006):ヘッド昇温エラーが発生しました。（同期メソッド実行時のみ）
その他	**ErrorCode 一覧を参照してください。**

Open メソッド

形式

void Open ();

説明

デバイスをオープンするときに呼び出します。

PosExplorer で指定したデバイスをオープンします。

Open メソッドが成功すると、共通プロパティ、その他のクラス固有プロパティが初期化されます。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
Illegal(106)	該当のコントロールはすでにオープンしています。

Release メソッド

形式

void Release ();

説明

デバイスの排他アクセスを解除するときにこのメソッドを呼び出します。

DeviceEnabled プロパティが **TRUE** で排他デバイスならば、デバイスはディセーブルにされます。イベント処理中（イベントハンドラ内）では実行しないでください。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
例外なし	排他アクセスが解除されました。**Claimed** プロパティは **FALSE** になります。
Illegal(106)	アプリケーションは該当のデバイスへの排他アクセス権を持っていません。
Busy	非同期処理が実行中です。
その他	**ErrorCode 一覧を参照してください。**

ResetStatistics メソッド

形式

void ResetStatistics ();

説明

このメソッドはサポートされていません。

例外

値	意味
Illegal(106)	このメソッドはサポートされません。

RetrieveStatistics メソッド

形式

sting RetrieveStatistics ();

説明

このメソッドはサポートされていません。

例外

値	意味
Illegal(106)	このメソッドはサポートされません。

形式

string RetrieveStatistics (StatisticCategories *statistics*);

説明

このメソッドはサポートされていません。

例外

値	意味
Illegal(106)	このメソッドはサポートされません。

形式

string RetrieveStatistics (string [] *statistics*);

説明

このメソッドはサポートされていません。

例外

値	意味
Illegal(106)	このメソッドはサポートされません。

UpdateFirmware メソッド

形式

void UpdateFirmware (String *FirmWareFileName*);

パラメータ	説明
FirmWareFileName	デバイスにダウンロードされるファームウェアを含むファイルか、ファームウェアファイルのセットを含むファイルの名称を指定します。

説明

このメソッドは、ファームウェアのバージョンがすでに POS プリンターのファームウェアのバージョンと同じか古いか新しいかに関わらず、*FirmWareFileName* パラメータで示されるファイルに含まれるファームウェアのバージョンに更新します。

このメソッドが起動されると、サービスオブジェクトは、指定されたファームウェアが存在していれば、このメソッドは直ちに終了し、ファームウェア更新処理の残りは非同期に継続されます。サービスオブジェクトは、OPOS_SUE_UF_PROGRESS(2100)に 1 から 100 を加えたファームウェア更新経過のパーセント値で **StatusUpdateEvent** を通知し、ファームウェア更新処理の状態をアプリケーションに報告します。アプリケーションが使いやすいように、**StatusUpdateEvent** の OPOS_SUE_UF_COMPLETE (2200) の値は OPOS_SUE_UF_PROGRESS(2100)+100 と同じ値に定義されています。ファームウェア更新を非同期に実行している途中でエラーが検出されれば、以下の **StatusUpdateEvent** が通知されます。

POS プリンターにファームウェアをダウンロード後、ファイル名から取得したファームウェアバージョンと、POS プリンターから取得したバージョンを比較し(**CompareFirmware** メソッドと同様の処理)、一致しない場合は OPOS_SUE_UF_COMPLETE(2200)は通知されず、OPOS_SUE_UF_FAILED_DEV_OK(2201)が通知されます。

※本機能を使用する場合には、AsyncMode 及び TransactionPrint, PageModePrint は指定しないでください。

値	意味
OPOS_SUE_UF_FAILED_DEV_OK(2201)	ファームウェア更新処理が失敗しましたが、デバイスはまだ操作可能です。

例外

このメソッドを呼び出したときに、次の PosControlException **をスローすることがあります。**

値	意味
例外なし	メソッドの実行に成功しました。
Noexist(109)	*FirmwareFileName* によって示されるファイルが存在しません。
Extended(114)	**ErrorCodeExtended**=OPOS_EFIRMWARE_BAD_FILE(281)：指定されたファームウェアファイルが正しくないかフォーマットか壊れています。（拡張子が "hx1" 以外の場合、この例外が発生します）

UpdateStatistics メソッド

形式

void UpdateStatistics (Statistic [] *statistics*);

説明

本メソッドはサポートしません。

例外

次の値のいずれかが戻され、**ErrorCode** プロパティにも格納されます。

値	意味
Illegal(106)	このメソッドはサポートされません。
Extended(114)	このメソッドはサポートされません。

形式

void UpdateStatistics (StatisticCategories *statistics*, object *value*);

説明

本メソッドはサポートしません。

例外

次の値のいずれかが戻され、**ErrorCode** プロパティにも格納されます。

値	意味
Illegal(106)	このメソッドはサポートされません。
Extended(114)	このメソッドはサポートされません。

4．5．専用プロパティ

AsyncMode プロパティ R/W

形式

bool AsyncMode;

説明

TRUE: **PrintNormal, CutPaper, PrintBarCode, PrintBitmap, RotatePrint, TransactionPrint,**
PageModePrint の印刷メソッドが非同期で実行されます。
FALSE: メソッドが同期で実行されます。
このプロパティは、**Open** メソッドにより **FALSE** に初期化されます。

CapCharacterSet プロパティ

形式

CharacterSetCapability CapCharacterSet;

説明

POS プリンターの印刷可能な文字設定を示します。
このプロパティには、次に示す値が入ります。

値	意味
Kanji(11)	文字設定は、コードページ 932 をサポートします。それは、0xA1 と 0xDF の間の半角のカタカナ、0x20 と 0x7F の間の全ての ASCII 文字をサポートします。また、JIS 第一水準～第三水準レベルで定義される文字のシフト JIS コード文字をサポートします。

このプロパティは、**Open** メソッドにより初期化されます。

Korean / Thai / Simplified Chinese / Traditional Chinese のいずれかの言語が選択されている場合は、シフト JIS コード文字をサポートしません。

Korean : KS コード文字をサポートします。
Thai : TIS コード文字をサポートします。
Simplified Chinese : GB2312 コード文字をサポートします。
Traditional Chinese : BIG5 コード文字をサポートします。

CapCoverSensor プロパティ

形式

bool CapCoverSensor;

説明

TRUE：POS プリンターは「カバーオープンセンサ」を持ちます。
このプロパティは、**Open** メソッドにより初期化されます。

CapMapCharacterSet プロパティ

形式

bool CapMapCharacterSet;

説明

TRUE：サービスオブジェクトは文字を **CharacterSetList** プロパティに定義された文字セットにマッピングすることが可能です。

CapRec2Color プロパティ

形式

bool CapRec2Color;

説明

TRUE：レシートの２色印字が可能です。（※設定プログラムで印字色を“Mono”設定している場合は、**FALSE**：レシートの 2 色印字は不可、が設定されます。）
このプロパティは、**Open** メソッドにより初期化されます。

CapRecBarCode プロパティ

形式

bool CapRecBarCode;

説明

TRUE：レシートのバーコード印刷が可能です。
このプロパティは、**Open** メソッドにより初期化されます。

CapRecBitmap プロパティ

形式

bool CapRecBitmap;

説明

TRUE：レシートのビットマップ印刷が可能です。
このプロパティは、**Open** メソッドにより初期化されます。

CapRecBold プロパティ

形式

bool CapRecBold;

説明

TRUE：レシートのボールド属性が可能です。

このプロパティは、**Open** メソッドにより初期化されます。

CapRecCartridgeSensor プロパティ

形式

PrinterCartridgeSensor CapRecCartridgeSensor;

説明

0：レシートカートリッジ センサの機能はサポートされません。

このプロパティは、**Open** メソッドにより初期化されます。

CapRecColor プロパティ

形式

PrinterColors CapRecColor;

説明

0：レシートカラー印刷の機能はサポートされません。

このプロパティは、**Open** メソッドにより初期化されます。

CapRecDhigh プロパティ

形式

bool CapRecDhigh;

説明

TRUE：レシートの縦倍角属性が可能です。

このプロパティは、**Open** メソッドにより初期化されます。

CapRecDwide プロパティ

形式

bool CapRecDwide;

説明

TRUE：レシートの横倍角属性が可能です。

このプロパティは、**Open** メソッドにより初期化されます。

CapRecDwideDhigh プロパティ

形式

bool CapRecDwideDhigh;

説明

TRUE：レシートの縦横倍角属性が可能です。

このプロパティは、**Open** メソッドにより初期化されます。

CapRecEmptySensor プロパティ

形式

bool CapRecEmptySensor;

説明

TRUE：レシートの「用紙無しセンサ」を持ちます。

このプロパティは、**Open** メソッドにより初期化されます。

CapRecItalic プロパティ

形式

bool CapRecItalic;

説明

FALSE：レシートのイタリック属性は不可能です。

このプロパティは、**Open** メソッドにより初期化されます。

CapRecLeft90 プロパティ

形式

bool CapRecLeft90;

説明

TRUE：レシートの左 90 度回転属性は可能です。

このプロパティは、**Open** メソッドにより初期化されます。

CapRecMarkFeed プロパティ

形式

PrinterMarkFeeds CapRecMarkFeed;

説明

0：マーク付き用紙の制御機能はサポートされていません。

このプロパティは、**Open** メソッドにより初期化されます。

CapRecNearEndSensor プロパティ

形式

bool CapRecNearEndSensor;

説明

TRUE：レシートの「用紙ニアエンドセンサ」を持ちます。

FALSE：レシートの「用紙ニアエンドセンサ」は機能しません。

このプロパティは、**Open** メソッドにより初期化され、設定プログラムによる設定 PNESense ＝ Enabled 設定の場合は **TRUE**, Disable 設定の場合は、**FALSE** が設定されます。

CapRecPageMode プロパティ

形式

bool CapRecPageMode;

説明

TRUE：レシートステーションにページモード機能があります。

このプロパティは、**Open** メソッドにより初期化されます。

CapRecPapercut プロパティ

形式

bool CapRecPapercut;

説明

TRUE：レシートの用紙カット機能が可能です。

このプロパティは、**Open** メソッドにより初期化されます。

CapRecPresent プロパティ

形式

bool CapRecPresent;

説明

TRUE：レシートが可能です。

このプロパティは、**Open** メソッドにより初期化されます。

CapRecRight90 プロパティ

形式

bool CapRecRight90;

説明

TRUE：レシートの右 90 度回転属性が可能です。

このプロパティは、**Open** メソッドにより初期化されます。

CapRecRotate180 プロパティ

形式

bool CapRecRotate180;

説明

TRUE：レシートの 180 度回転属性が可能です。
このプロパティは、**Open** メソッドにより初期化されます。

CapRecStamp プロパティ

形式

bool CapRecStamp;

説明

FALSE：レシートのスタンプ印刷が不可能です。
このプロパティは、**Open** メソッドにより初期化されます。

CapRecUnderline プロパティ

形式

bool CapRecUnderline;

説明

TRUE：レシートのアンダーライン属性が可能です。
このプロパティは、**Open** メソッドにより初期化されます。

CapTransaction プロパティ

形式

bool CapTransaction;

説明

TRUE：POS プリンターの一括処理が有効です。
このプロパティは、**Open** メソッドにより初期化されます。

CartridgeNotify プロパティ R/W

形式

PrinterCartridgeNotify CartridgeNotify;

説明

カートリッジの状態通知機能の有効/無効を示します。

このプロパティは、アプリケーションにより設定されます。

値	意味
Disabled(0)	コントロールは、アプリケーションに対してカートリッジ状態通知を提供しません。カートリッジ状態通知に関する **StatusUpdateEvent** は通知されず、**JrnCartridgeState**、**RecCartridgeState**、**SlpCartridgeState** プロパティには何も設定されません。

このプロパティは、**Open** メソッドで Disabled(0) に初期化されます。

例外

このメソッドを呼び出したときに、次の PosControlException **をスローすることがあります。**

値	意味
Illegal(106)	本プロパティの設定はできません。
その他	ErrorCode 一覧を参照してください。

CharacterSet プロパティ R/W

形式

int CharacterSet;

説明

文字の印字に用いるキャラクタを設定します。

このプロパティは、**Open** メソッド後、最初にデバイスをイネーブルにした時に初期化されます。

このプロパティには、次に示す値の内の一つが設定されます。

< 言語が English(Latin) または Japanese の場合 >

値	意味
437	PC437(USA:Standard Europe)文字セットを選択します。
850	PC850(Multilingual)文字セットを選択します。
852	PC852(Latin2)文字セットを選択します。
857	PC857(Turkish)文字セットを選択します。
858	PC858(Euro)文字セットを選択します。
860	PC860 (Portuguese)文字セットを選択します。
863	PC863(Candian-French)文字セットを選択します。
864	PC864 (Arabic without BOX DRAWINGS below 20) 文字セットを選択します。
865	PC865(Nordic)文字セットを選択します。
866	PC866 文字セットを選択します。
869	PC869(Greece)文字セットを選択します。
932	Windows のコードページ；日本語 Shift-JIS。
PTR_CS_ASCII (998)	ASCII キャラクタを設定します。0x20 と 0x7F の間の ASCII キャラクタをサポートします。この定数値は 998 です。
1252	WPC1252 文字セットを選択します。
28592	ISO8859-2(1999 Latin Alphabet No.2)文字セットを選択します。
28597	ISO8859-7(1987 LatinGreek Alphabet)文字セットを選択します。

< 言語が Korean / Thai / Simplified Chinese / Traditional Chinese の場合>

値	意味
437	PC437(USA:Standard Europe)文字セットを選択します。
850	PC850(Multilingual)文字セットを選択します。
858	PC858(Euro)文字セットを選択します。
860	PC860 (Portuguese)文字セットを選択します。
863	PC863(Candian-French)文字セットを選択します。
865	PC865(Nordic)文字セットを選択します。
874	タイ語 Windows コードページ（TIS コード）を選択します。 (言語が Thai の場合のみ)
936	簡体中国語 Windows コードページ(GB2312)を選択します。 (言語が Simplified Chinese の場合のみ)
949	韓国語 Windows コードページ（KS コード）を選択します。 (言語が Korean の場合のみ)
950	繁体中国語 Windows コードページ（Big5 + HKSCS）を選択します。 (言語が Traditional Chinese の場合のみ)
PTR_CS_ASCII (998)	ASCII キャラクタを設定します。0x20 と 0x7F の間の ASCII キャラクタをサポートします。この定数値は 998 です。

例外

このメソッドを呼び出したときに、次の PosControlException **をスローすることがあります。**

値	**意味**
例外なし	このプロパティの設定に成功しました。
Illegal(106)	不正な値が使用されました。
その他	**ErrorCode 一覧を参照してください。**

CharacterSetList プロパティ

形式

int[] CharacterSetList;

説明

キャラクタ設定番号の文字列。
言語が English(Latin) または Japanese の場合、
"437,850,852,857,858,860,863,864,865,866,869,932,998,1252,28592,28597" が設定されます。
言語が Korean の場合、"437,850,858,860,863,865,932,949,998" が設定されます。
言語が Thai language の場合、"437,850,858,860,863,865,874,932,998" が設定されます。
言語が Simplified Chinese の場合、"437,850,858,860,863,865,932,936,998" が設定されます。
言語が Traditional Chinese の場合、"437,850,858,860,863,865,932,950,998" が設定されます。

インストーラにより、この値のいずれかが設定されます。このプロパティは、**Open** メソッドにより初期化されます。

CoverOpen プロパティ

形式

bool CoverOpen;

説明

TRUE：POS プリンターのカバーが開いています。
FALSE：閉まっています。
このプロパティは、デバイスをイネーブルにしている間に初期化され、現状を保持します。

ErrorLevel プロパティ

形式

PrinterErrorLevel ErrorLevel;

説明

エラー状態の重大さを示します。

このプロパティには、次に示す値の内の一つが設定されます。

値	意味
None(1)	エラー状態ではありません。
Recoverable(2)	復帰可能エラーが発生しました。（カバーオープン、レシートエンド、ヘッド昇温、カッタージャムエラー発生時、電源断時）
Fatal(3)	復帰不可能エラーが発生しました。（致命的エラーのみ）

このプロパティは、**ErrorEvent** が通知される前に、コントロールにより設定されます。エラーが解除されれば、このプロパティは None(1)に変わります。

ErrorStation プロパティ

形式

PrinterStation ErrorStation;

説明

エラーを察知した時に、印刷している POS プリンター（Receipt(2)）を保持します。

このプロパティは **ErrorEvent** が通知される前に設定されます。

電源オフ（またはケーブル取り外し）では"0"がセットされます。

ErrorString プロパティ

形式

String ErrorString;

説明

現在のエラーについてのベンダー固有の記述を保持します。

このプロパティは、**ErrorEvent** が通知される前に、コントロールにより設定されます。この記述を利用しないならば、プロパティには空文字列が設定されます。エラーが解除されれば、このプロパティは空文字列に変わります。

POS プリンターで設定されるのは以下の文言です。

・カバーオープン発生時　"Cover Open "
・レシートエンド発生時　"Paper End"
・ヘッド昇温発生時　"Head Hot"
・致命的エラー発生時　"Fatal Error”
・カッタージャムエラー発生時　"Cutter Jam Error"
・電源断（オフライン）時　"Power Off or Offline”

FlagWhenIdle プロパティ R/W

形式

bool FlagWhenIdle;

説明

TRUE：POS プリンターコントロールがアイドル状態であれば、**StatusUpdateEvent** を通知します。

FALSE：このイベントは通知されません。

このステータスイベントが通知されると、**FlagWhenIdle** は自動的に **FALSE** にリセットされます。

このプロパティを使用してステータスイベントを利用することにより、アプリケーションは全ての非同期出力が終了した時を知ることができます。出力が正常終了した時、または **ErrorEvent** を受け取ったイベントハンドラにより出力が削除された時に、イベントが通知されます。

もし、**FlagWhenIdle** プロパティを **TRUE** に設定する時に、**State** プロパティが既に Idle(2)であれば、**StatusUpdateEvent** は直ちに通知されます。従って、非同期出力の終了とこのフラグの設定のすれ違いを気にせずに、アプリケーションはこのイベントを使用できます。

このプロパティは、**Open** メソッドにより **FALSE** に初期化されます。

本プロパティ設定時、次の値が **ErrorCode** プロパティに格納されます。

値	意味
OPOS_SUCCESS(0)	プロパティは正常に設定されました。

FontTypefaceList プロパティ

形式

string[] FontTypefaceList;

説明

空文字列が設定されています。デフォルトのフォントタイプだけがサポートされていることを示します。

このプロパティは、**Open** メソッドにより初期化されます。

MapCharacterSet プロパティ R/W

形式

bool MapCharacterSet;

説明

TRUE : 印字の際にサービスオブジェクトはアプリケーションから渡された文字を **CharacterSet** プロパティで選択された文字セットにマップします。

このプロパティは、**Open** メソッドにより **TRUE** に初期化されます。

MapMode プロパティ R/W

形式

MapMode MapMode;

説明

POS プリンターのマッピングモードを示します。マッピングモードは他のプロパティで使用されるラインの高さや行間を示すような尺度の単位を定義します。

以下のマップモードをサポートしています。()内の値は、単位あたりを dot 換算した値です。

値	意味
Dots(1)	POS プリンターのドット幅 0.125mm (1 dot)
Twips(2)	1 インチの 1/1440 （7.0866 dot）
English(3)	0.001 インチ （4.921 dot）
Metric(4)	0.01 ミリメータ (12.5 dot)

MapMode を設定すると、**RecLineSpacing**、**RecLineWidth**、**RecLineHeight** も変化します。

Open メソッド後最初にデバイスをイネーブルにした時に、Dots(1)に初期化されます。

例外

このメソッドを呼び出したときに、次の PosControlException **をスローすることがあります。**

値	意味
例外なし	プロパティは正常に設定されました。
Illegal(106)	不正なマッピングモードが指定されました。

PageModeArea プロパティ

形式

System.Drawing.Point PageModeArea;

説明

PageModeStation プロパティに指定したステーションのページモード領域を、**MapMode** プロパティで指定した単位で保持します。ページモード領域は、ページモード印刷領域とは異なり、printer のハードウエア的な能力で決定されます。このプロパティは、ページモード領域の幅と高さを表すカンマ区切りの 2 つの ASCII 数字で構成されます。

例えば、文字列が"450,800"であれば、ページモード領域は幅 450 単位、高さ 800 単位です。そのステーションのページモード領域は、左上隅(0,0)と右下隅(449,799)で囲まれる長方形で示されます。

PageModeDescriptor プロパティ

形式

PageModeDescriptors PageModeDescriptor;

説明

PageModeStation プロパティに指定したステーションで利用可能なページモード機能を、下記の値の論理和で示します。

本クラスライブラリでは下記値の論理和が初期値となります。

値	意味
Bitmap(1)	ビットマップ印刷
Barcode(2)	バーコード印刷
BitmapRotate(4)	ビットマップの回転印刷
BarcodeRotate(8)	バーコードの回転印刷

PageModeHorizontalPosition プロパティ R/W

形式

int PageModeHorizontalPosition;

説明

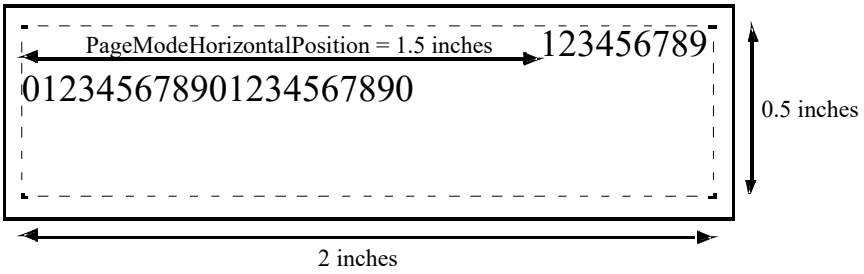
PageModeStation に指定したステーションで、ページモード印刷領域内の印刷開始位置を水平方向に絶対位置で指定するための値です。
MapMode プロパティに指定した単位で表記します。

水平方向とは、**PageModePrintDirection** プロパティで設定した印刷方向と同じ方向を指します。
値を小刻みに指定することができないデバイスの場合、それに最も近い位置から印刷を開始します。
このプロパティは、指定された絶対位置を保持します。

PageModeHorizontalPosition プロパティの使い方を、以下のコード記述例で示します。

```
myptr.setMapMode(English);
myptr.setPageModeStation(Receipt);
myptr.pageModePrint(PageMode);
// Set print area to 2 inches by 0.5 inches
myptr.setPageModePrintArea("0,0,2000,500");
myptr.setPageModePrintDirection(LeftToRight);
myptr.setPageModeHorizontalPosition(1500);
myptr.printNormal(Receipt, "1234567890123456789012345678 90¥n");
```

上記のコードを記述した場合、下記のレシート印刷結果となります。



PageModePrintArea プロパティ R/W

形式

System.Drawing.Rectangle PageModePrintArea;

説明

PageModeStation に指定したステーションのページモード印刷領域を、**MapMode** プロパティで指定した単位で保持します。ページモード印刷領域のサイズは、ページモード領域より大きく取ることはできません。

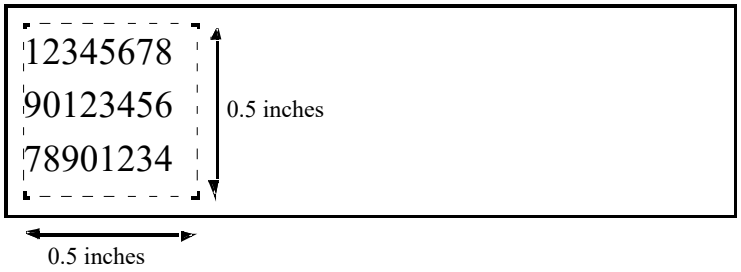
このプロパティは、カンマ区切りの４つの ASCII 数字で構成されます。始点の水平方向座標、始点の垂直方向座標、水平方向の幅、垂直方向の高さ、の順に列挙します。
例えば文字列が、”50,100,200,400”である場合、そのステーションのページモード印刷領域は、左上隅(50,100)と右下隅(249,499)で囲まれる長方形で示されます。
このプロパティは、”0,0,0,0”に初期化されます。

ページモード印刷領域の右端を超える文字は、次の行に印刷されます。
ページモード印刷領域の下端を超える文字と画像は、印刷されません。

具体例を示します。

```
myptr.setMapMode(English);
myptr.setPageModeStation(Receipt);
myptr.pageModePrint(PageMode);
// Set print area to half inch square block
myptr.setPageModePrintArea(“0,0,500,500”);
myptr.setPageModePrintDirection(LeftToRight);
myptr.printNormal(Receipt,“1234567890123456789012345678 90¥n”);
```

上記のコードを記述した場合、下記のレシート印刷結果となります。



PageModePrintDirection プロパティ R/W

形式

PageModePrintDirection PageModePrintDirection;

説明

ページモード印刷領域の印刷方向を保持します。印刷方向は以下です。

値	意味
LeftToRight(1)	左上隅を始点に、左から右方向へ印刷します。（通常の印刷方向）
BottomToTop(2)	左下隅を始点に、下から上方向へ印刷します。（左 90 度の回転印刷）
RightToLeft(3)	右下隅を始点に、右から左方向へ印刷します。（180 度の回転印刷）
TopToBottom(4)	右上隅を始点に、上から下方向へ印刷します。（右 90 度の回転印刷）

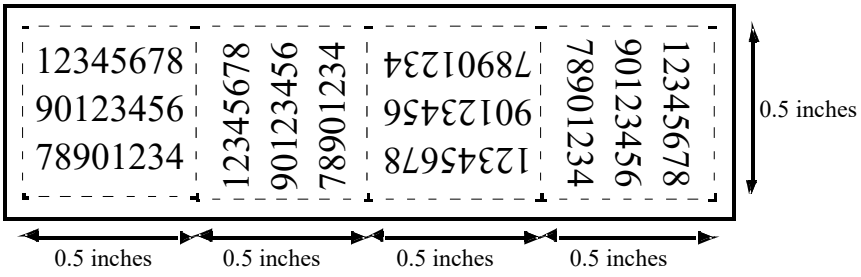
このプロパティは Open メソッド実行後、最初にイネーブルされた時点で、LeftToRight に初期化されます。

このプロパティを変更すると、**PageModeHorizontalPosition** プロパティと **PageModeVerticalPosition** プロパティで示される印刷開始点の補正方向も変化します。このプロパティの変更は、現在編集中のページモード印刷領域に対してのみ効果があります。
ページモード印刷領域を切り替えることで、文字の回転方向を組み合わせたレシートや単票を印刷することができます。

具体例を示します。

```
myptr.setMapMode(English);
myptr.setPageModeStation(Receipt);
myptr.pageModePrint(PageMode);
// Set print area to half inch square block
myptr.setPageModePrintArea("0,0,500,500");
myptr.setPageModePrintDirection(LeftToRight);
myptr.printNormal(Receipt,"1234567890123456789012345678 90¥n");
myptr.setPageModePrintArea("500,0,500,500");
myptr.setPageModePrintDirection(TopTpBottom);
myptr.printNormal(Receipt,"1234567890123456789012345678 90¥n");
myptr.setPageModePrintArea("1000,0,500,500");
myptr.setPageModePrintDirection(LeftToRight);
myptr.printNormal(Receipt,"1234567890123456789012345678 90¥n");
myptr.setPageModePrintArea("1500,0,500,500");
myptr.setPageModePrintDirection(TopTpBottom);
myptr.printNormal(Receipt,"1234567890123456789012345678 90¥n");
```

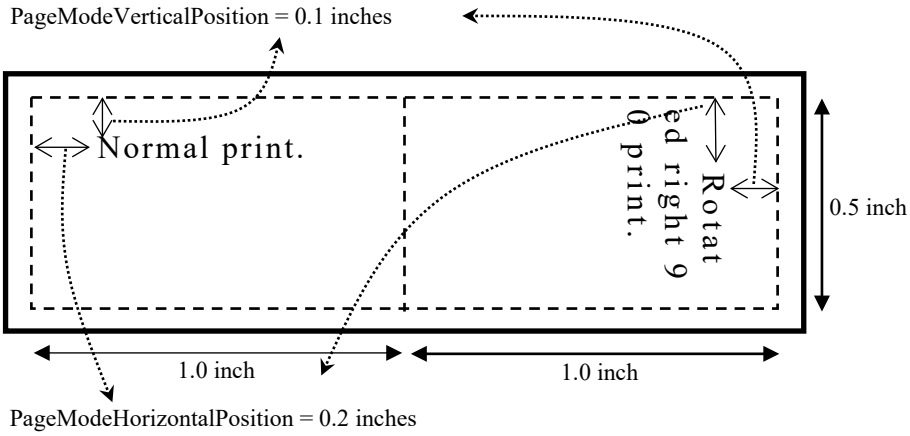
上記のコードを記述した場合、下記のレシート印刷結果となります。



同様に文字を回転させることもできます。

```
myptr.setMapMode(English);
myptr.setPageModeStation(Receipt);
myptr.pageModePrint(PageMode);
myptr.pageModeVerticalPosition(100);
myptr.pageModeHorizontalPosition(200);
myptr.setPageModePrintArea("0,0,1000,500");
myptr.setPageModePrintDirection(LeftToRight);
myptr.printNormal(Receipt, "Normal print.¥n");
myptr.setPageModePrintArea("1000,0,1000,500");
myptr.setPageModePrintDirection(TopToBottom);
myptr.printNormal(Receipt, "Rotated right 90 print.¥n");
myptr.setPageModePrint(Normal);
```

上記のコードを記述した場合、下記のレシート印刷結果となります。



PageModeStation プロパティ R/W

形式

PrinterStation PageModeStation;

説明

ページモード機能の対象とするステーションを指定します。
本クラスライブラリでは Receipt(2)のみ指定可能です。

PageModeVerticalPosition プロパティ R/W

形式

int PageModeVerticalPosition;

説明

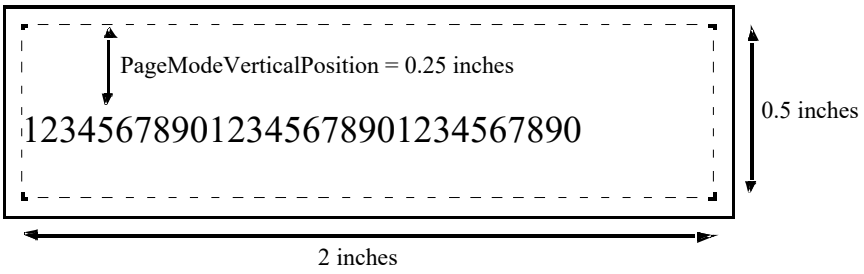
PageModeStation に指定したステーションで、ページモード印刷領域内の印刷開始位置を垂直方向に絶対位置で指定するための値です。
MapMode プロパティに指定した単位で表記します。

垂直方向とは、**PageModePrintDirection** プロパティで設定した印刷方向に対して垂直な方向です。
値を小刻みに指定することができないデバイスの場合、それに最も近い位置から印刷を開始します。
このプロパティは、現在位置ではなくて、最後に指定した垂直方向の絶対位置の値を返します。

PageModeVerticalPosition プロパティの使い方を、以下のコード記述例で示します。

```
myptr.setMapMode(English);
myptr.setPageModeStation(Receipt);
myptr.pageModePrint(PageMode);
// Set print area to 2 inches by 0.5 inches
myptr.setPageModePrintArea("0,0,2000,500");
myptr.setPageModePrintDirection(LeftToRight);
myptr.setPageModeVerticalPosition(250);
myptr.printNormal(Receipt,"1234567890123456789012345678 90¥n");
```

上記のコードを記述した場合、下記のレシート印刷結果となります。



RecBarCodeRotationList プロパティ

形式

Rotation[] RecBarCodeRotationList;

説明

この文字列は、レシートのバーコードが回転できる方向を示します。

"0,R90,L90,180"が設定されます。

このプロパティは、**Open** メソッドにより初期化されます。文字列は、カンマで区切られた回転方向を示す文字列群で構成されます。回転方向を示す文字列は以下の内容になります。

値	意味
0	バーコードは通常の方向に印刷できます。
R90	バーコードは右 90 度に回転印刷できます。
L90	バーコードは左 90 度に回転印刷できます。
180	バーコードは 180 度(倒立)に回転印刷できます。

RecBitmapRotationList プロパティ

形式

Rotation[] RecBitmapRotationList;

説明

この文字列は、レシートのビットマップが回転できる方向を示します。

このプロパティは、**Open** メソッドにより初期化されます。文字列は、カンマで区切られた回転方向を示す文字列群で構成されます。回転方向を示す文字列は以下の内容になります。

値	意味
0	ビットマップバーコードは通常の方向に印刷できます。
R90	ビットマップバーコードは右 90 度に回転印刷できます。
L90	ビットマップバーコードは左 90 度に回転印刷できます。
180	ビットマップバーコードは 180 度(倒立)に回転印刷できます。

RecCartridgeState プロパティ

形式

PrinterCartridgeStates RecCartridgeState;

説明

現在選択されているレシートのカートリッジ（インク、リボン、トナー）の状態を示します。

本 POS プリンターはサーマルプリンターなので、この値は以下の値で固定値です。

値	意味
Unknown(268435456)	デバイスは、カートリッジ状態通知機能をサポートしていません。

このプロパティは、デバイスをイネーブルにしている間に初期化され、現状を保持します。

RecCurrentCartridge プロパティ R/W

形式

PrinterColors RecCurrentCartridge;

説明

レシートカートリッジの選択はサポートされません。0 で初期化されています。

例外

このメソッドを呼び出したときに、次の PosControlException **をスローすることがあります。**

値	意味
Illegal(106)	カートリッジの指定は無効です。
その他	**ErrorCode 一覧を参照してください。**

RecEmpty プロパティ

形式

bool RecEmpty;

説明

TRUE：レシート用紙は紙切れです。

FALSE：レシート用紙はあります。

このプロパティは、デバイスをイネーブルにしている間に初期化され、現状を保持します。

RecLetterQuality プロパティ R/W

形式

bool RecLetterQuality;

説明

TRUE：高品質印字モードによる印刷になります。
FALSE：標準印字モードによる印刷になります。

このプロパティは、**Open** メソッド後最初にデバイスをイネーブルにした時に、**TRUE** に初期化されます。
高品質印字モードの影響の対象は、内蔵文字およびダウンロード文字で、且つ縦横いずれも 2 倍以上の場合、スムージング処理された印字結果が得られますが、標準印字モードに比べ、印字スピードが若干低下します。
標準印字モードの場合、ビットマップは、1/3 の解像度で印刷されます（出力される大きさは高品質モードと同じですが、解像度が 1/3 となります。）。また、縦横 2 倍以上の内蔵文字を印字させる場合においても、スムージング処理は行われません。
また、**SetBitmap** でビットマップを登録する場合は、その時点での **RecLetterQuality** に左右されません。
エスケープシーケンスにおけるビットマップ印字を行う際に、高品質印字モードの場合は通常解像度で、標準印字モードの場合は 1/2 倍の解像度(大きさは縦横 2 倍)で印刷されます。**PrintBitmap** メソッドも同様に **RecLetterQuality** に従います。
(※PosExplorer 外部設定値で、Smoothing 設定を off にした場合、**TRUE** 設定時もフォントに関しては、スムージング処理を行いません。)

RecLineChars プロパティ R/W

形式

int RecLineChars;

説明

レシートの一行に印字される半角文字の文字数です。

指定された桁数により、以下のフォントで印字されます。

180dpi モードが無効な場合、以下の通りです。

桁数	印字フォント(横×縦)
プリンター設定：用紙幅 80mm(576 ドット)：	
48（全角 24）	12×24 ドット(フォント A)
57（全角 28）	10×24 ドット(フォント B)
72（全角 36）	8×16 ドット(フォント C) *1
プリンター設定：用紙幅 80mm(512 ドット)：	
42（全角 21）	12×24 ドット(フォント A)
51（全角 25）	10×24 ドット(フォント B)
64（全角 32）	8×16 ドット(フォント C) *1
プリンター設定：用紙幅 58mm(420 ドット)：	
35（全角 17）	12×24 ドット(フォント A)
42（全角 21）	10×24 ドット(フォント B)
52（全角 26）	8×16 ドット(フォント C) *1
プリンター設定：用紙幅 58mm(384 ドット)：	
32（全角 16）	12×24 ドット(フォント A)
38（全角 19）	10×24 ドット(フォント B)
48（全角 24）	8×16 ドット(フォント C) *1
プリンター設定：用紙幅 50mm(360 ドット)：	
30（全角 15）	12×24 ドット(フォント A)
36（全角 18）	10×24 ドット(フォント B)
45（全角 22）	8×16 ドット(フォント C) *1

*1： 言語が English(Latin)または Japanese の場合のみ有効。

180dpi モードが有効の場合は、以下の通りです。

桁数	印字フォント(横×縦)
プリンター設定：用紙幅 80mm(512 ドット)：	
42（全角 21）	12×24 ドット(フォント A)
51（全角 25）	10×24 ドット(フォント B)
64（全角 32）	8×16 ドット(フォント C) *1
プリンター設定：用紙幅 58mm(360 ドット)：	
30（全角 15）	12×24 ドット(フォント A)
36（全角 18）	10×24 ドット(フォント B)
45（全角 22）	8×16 ドット(フォント C) *1

*1： 言語が English(Latin)または Japanese の場合のみ有効。

この値が、サポートされている行文字幅に変われば、文字幅はその指定された値に設定されます。正確な幅がサポートできない場合、サポートされている行文字幅の中で、指定された値より大きく、かつ最も近い値に設定されます。（例えば、用紙幅 83mm 設定時に、プリンターに 40 をセットした場合、サービスオブジェクトは文字のサイズ「53」を選択します）。もし、文字幅がサポートできなければ、例外が返されます。

RecLineChars を設定することにより、**RecLineHeight, RecLineSpacing, RecSideWayMaxChars, RecSidewaysMaxlines** プロパティも更新されます。

例外

このメソッドを呼び出したときに、次の PosControlException **をスローすることがあります。**

値	意味
例外なし	プロパティは正常に設定されました。
Illegal(106)	不正な行文字幅が指定されました。

RecLineCharsList プロパティ

形式

int[] RecLineCharsList;

説明

レシートによりサポートされる行文字幅を含む文字列です。

このプロパティは、**Open** メソッドにより、プリンターの用紙幅、設定によって以下の値に初期化されます。

180dpi モードが無効な場合、以下の通りです。

プリンター用紙幅	値
用紙幅 80mm(576 ドット)	“48,57,72(*)”
用紙幅 80mm(512 ドット)	“42,51,64(*)”
用紙幅 58mm(420 ドット)	“35,42,52(*)”
用紙幅 58mm(384 ドット)	“32,38,48(*)”
用紙幅 50mm(360 ドット)	“30,36,45(*)”

* 言語が English(Latin) または Japanese の場合のみ有効。

180dpi モードが有効な場合は、以下の通りです。

プリンター用紙幅	値
用紙幅 80mm(512 ドット)	“42,51,64(*)”
用紙幅 58mm(360 ドット)	“30,36,45(*)”

* 言語が English(Latin) または Japanese の場合のみ有効。

RecLineHeight プロパティ R/W

形式

int RecLineHeight;

説明

レシート印刷の高さです。**MapMode** で定義された単位で表記します。

RecLineChars が変わると、**RecLineHeight** は、選択された幅に対するデフォルトの行高に更新されます。

RecLineHeight の値は、**Open** メソッドにより POS プリンターのデフォルトの行高に初期化されます。

取り得る値は、180dpi モードが無効の場合、以下の通りです。（※プロパティの値は、**MapMode** プロパティが Dots(1)の値）

桁数	RecLineHeight プロパティの値
プリンター設定：用紙幅 80mm(576 ドット)：	
48	24
57	24
72	16 *1
プリンター設定：用紙幅 80mm(512 ドット)：	
42	24
51	24
64	16 *1
プリンター設定：用紙幅 58mm(420 ドット)：	
35	24
42	24
52	16 *1
プリンター設定：用紙幅 58mm(384 ドット)：	
32	24
38	24
48	16 *1
プリンター設定：用紙幅 50mm(360 ドット)：	
30	24
36	24
45	16 *1

*1： 言語が English(Latin)または Japanese の場合のみ有効。

180dpi モードが有効な場合は、以下の通りです。

桁数	RecLineHeight プロパティの値
プリンター設定：用紙幅 80mm(512 ドット)：	
42	24
51	24
64	16　*1
プリンター設定：用紙幅 58mm(360 ドット)：	
30	24
36	24
45	16　*1

*1： 言語が English(Latin)または Japanese の場合のみ有効。

例外

このメソッドを呼び出したときに、次の PosControlException **をスローすることがあります。**

値	意味
Illegal(106)	本プロパティの設定はできません。取得のみ可能です。

RecLineSpacing プロパティ R/W

形式

int RecLineSpacing;

説明

通常文字の印刷行の高さを示します。すなわち、印字行の高さと行間スペースの高さの両方を加えたものです。本プロパティは **MapMode** で定義された単位で表記します。

RecLineChars が変わったとき、新しい **RecLineHeight** が **RecLineSpacing** に指定された値より大きい場合は、**RecLineHeight** と同じ値が設定されます。

RecLineSpacing の値は、**Open** メソッド後に、POS プリンターのデフォルトの行間に初期化されます。

16(dot) ～ 127(dot) の範囲で設定が可能となります。

例外

このメソッドを呼び出したときに、次の PosControlException **をスローすることがあります。**

値	意味
例外なし	プロパティは正常に設定されました。
Illegal(106)	プロパティの設定範囲が不正です。
その他	**ErrorCode 一覧**を参照してください。

RecLinesToPaperCut プロパティ

形式

int RecLinesToPaperCut;

説明

レシート用紙が切られる前に、進まなければならない行数を保持します。

これは用紙カット機構に着くまでの行数です。

RecLineChars プロパティや **RecLineSpacing** プロパティを変えることによって、このプロパティは更新されます。

RecLineWidth プロパティ

形式

int RecLineWidth;

説明

RecLineChars の一行の幅です。**MapMode** で定義された単位で表記します。
このプロパティは、**Open** メソッド後に初期化されます。
180dpi モードが無効の場合、プリンターの用紙幅により、以下の値が設定されます。

プリンター用紙幅	値
用紙幅 80mm	576
用紙幅 80mm	512
用紙幅 58mm	420
用紙幅 58mm	384
用紙幅 50mm	360

180dpi モードが有効の場合は、以下の通りです。

プリンター用紙幅	値
用紙幅 80mm	512
用紙幅 58mm	360

RecNearEnd プロパティ

形式

bool RecNearEnd;

説明

TRUE：レシート用紙がニアエンドです。
FALSE：ニアエンドではありません。
このプロパティは、デバイスをイネーブルにした時に初期化され、イネーブルしている間、カレント値が保持されます。

RecSidewaysMaxChars プロパティ

形式

int RecSidewaysMaxChars;

説明

サイドウェイモード（左右 90 度回転印刷）において、各行に印刷される半角文字の最大文字数です。下記の値を取ります。CapRec2Color プロパティが TRUE の場合(2 色時)は、左右回転印刷の幅が 1/2 となってしまうため、印字可能な文字数は半分の値となります。

桁数	サイドウェイモード印字桁数	2 色時
プリンター設定：用紙幅 80mm(576 ドット)：		
48（全角 24）	138	69
57（全角 28）	166	83
72（全角 36）*1	207	103
プリンター設定：用紙幅 80mm(512 ドット)：		
42（全角 21）	138	69
51（全角 25）	166	83
64（全角 32）*1	207	103
プリンター設定：用紙幅 58mm(420 ドット)：		
35（全角 17）	138	69
42（全角 21）	166	83
52（全角 26）*1	207	103
プリンター設定：用紙幅 58mm(384 ドット)：		
32（全角 16）	138	69
38（全角 19）	166	83
48（全角 24）*1	207	103
プリンター設定：用紙幅 50mm(360 ドット)：		
30（全角 15）	138	69
36（全角 18）	166	83
45（全角 22）*1	207	103

*1： 言語が English(Latin)または Japanese の場合のみ有効。

RecSidewaysMaxLines プロパティ

形式

int RecSidewaysMaxLines;

説明

サイドウェイモード（左右 90 度回転印刷）において、印刷される最大行数です。

RecLineWidth プロパティを **RecLineSpacing** プロパティで割った値となります。ただし、割った値の余りが、**RecLineHeight** プロパティ（フォントの高さ）と等しいか、あるいは大きい場合は、割った値に +1 された値となります。従って **RecLineSpacing** プロパティを変えることにより、このプロパティは変わります。

ただし、フォント C(※**RecLineChars** プロパティを参照してください)の場合に限り、上記条件に対して、**RecLineWidth** − 7(dot)の値に対して計算が行なわれます。

本プロパティは、**Open** メソッド後最初にデバイスをイネーブルにした時に初期化されます。

RotateSpecial プロパティ R/W

形式

Rotation RotateSpecial;

説明

バーコードの回転方向を示します。

このプロパティは、 **Open** メソッドにより Normal(1)に初期化されます。

このプロパティには、次に示す値の内の一つが設定されます。

値	意味
Normal(1)	バーコードは通常の方向に印刷できます。
Right90(257)	バーコードは右 90 度に回転印刷できます。
Left90(258)	バーコードは左 90 度に回転印刷できます。
Rotate180(259)	バーコードは 180 度(倒立)に回転印刷できます。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
例外なし	プロパティは正常に設定されました。
Illegal(106)	不正なプロパティ値が指定されました。

※以下の POS プリンター専用プロパティはサポートされません。

bool CapConcurrentJrnRec;
bool CapConcurrentJrnSlp;
bool CapConcurrentRecSlp;
bool CapConcurrentPageMode;
bool CapJrn2Color;
bool CapJrnBold;
void CapJrnCartridgeSensor;
void CapJrnColor;
bool CapJrnDhigh;
bool CapJrnDwide;
bool CapJrnDwideDhigh;
bool CapJrnEmptySensor;
bool CapJrnItalic;
bool CapJrnNearEndSensor;
bool CapJrnPresent;
bool CapJrnUnderline;
bool CapSlp2Color;
bool CapSlpBarCode;
bool CapSlpBitmap;
bool CapSlpBold;
boolCapSlpBothSidesPrint;
void CapSlpCartridgeSensor;
void CapSlpColor;
bool CapSlpPageMode;
bool CapSlpDhigh;
bool CapSlpDwide;
bool CapSlpDwideDhigh;
bool CapSlpEmptySensor;
bool CapSlpFullslip;
bool CapSlpItalic;
bool CapSlpLeft90;
bool CapSlpNearEndSensor;
bool CapSlpPresent;
bool CapSlpRight90;
bool CapSlpRotate180;
bool CapSlpUnderline;

void JrnCartridgeState;
void JrnCurrentCartridge;
bool JrnEmpty;
bool JrnLetterQuality;
void JrnLineChars;
string JrnLineCharsList;
void JrnLineHeight;
void JrnLineSpacing;
void JrnLineWidth;
bool JrnNearEnd;
string SlpBarCodeRotationList;
string SlpBitmapRotationList;
void SlpCartridgeState;
void SlpCurrentCartridge;
bool SlpEmpty;
bool SlpLetterQuality;
void SlpLineChars;
string SlpLineCharsList;
void SlpLineHeight;
void SlpLinesNearEndToEnd;
void SlpLineSpacing;
void SlpLineWidth;
void SlpMaxLines;
bool SlpNearEnd;
void SlpSidewaysMaxChars;
void SlpSidewaysMaxLines;
void SlpPrintSide;

4．6．専用メソッド

BeginInsertion メソッド

形式

void BeginInsertion (int *Timeout*);

説明

このメソッドはスリッププリンターを対象としているため、本クラスライブラリではサポートされません。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
Illegal(106)	POS プリンターはスリップを持っていません。
その他	**ErrorCode 一覧を参照してください。**

BeginRemoval メソッド

形式

void BeginRemoval (int *Timeout*);

説明

このメソッドはスリッププリンターを対象としているため、本クラスライブラリではサポートされません。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
Illegal(106)	POS プリンターはスリップを持っていません。
その他	**ErrorCode 一覧を参照してください。**

ChangePrintSide メソッド

形式

void ChangePrintSide (PrinterSide *Side*);

説明

このメソッドはスリッププリンターを対象としているため、本クラスライブラリではサポートされません。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
Illegal(106)	POS プリンターはスリップを持っていません。
その他	**ErrorCode 一覧を参照してください。**

ClearPrintArea メソッド

形式

void ClearPrintArea ();

説明

PageModePrintArea プロパティで定義したページモード印刷領域上の印刷データを消去します。
ページモード領域全体を消去したい場合は、PageModeArea プロパティで示される領域と同等の領域を PageModePrintArea プロパティに指定し、Cancel を引数として PageModePrint メソッドを呼び出します。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
例外なし	メソッドは正常終了しました。
その他	**ErrorCode 一覧を参照してください。**

CutPaper メソッド

形式

void CutPaper (int *Percentage*);

Percentage パラメータは、カットすべき用紙のパーセンテージを示します。1～99 が設定された場合は"パーシャルカット"が、100 が設定された場合には"フルカット"を実行します。

1～100 以外が指定された場合は、Illegal(106) が返ります。

説明

レシート用紙を切る時にこのメソッドを呼び出します。

このメソッドは、**AsyncMode** が **FALSE** ならば、同期的に実行され、**AsyncMode** が **TRUE** ならば、非同期的に実行されます。**PrintNormal** や **PrintImmediate** メソッド呼び出しの際に、用紙カットのエスケープシーケンスを用いても用紙カットを行うことができます。尚、POS プリンターにバッファリングされているデータがある場合（印字要求をしたが、POS プリンターに印字されていない場合）は、カットすることができません。レシートカットを行うためには行の先頭である必要があります。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
例外なし	メソッドは正常終了しました。
Illegal(106)	不正なパーセンテージが指定されました。
NoHardware(107)	POS プリンターはオフ、またはオフラインです。
Failure(111)	本クラスライブラリはエラー状態です。エラー状態を解除して実行してください。
Busy(113)	出力中なので実行できません。
Extended(114)	**ErrorCodeExtended** = CoverOpen(201): POS プリンターカバーが開いています。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = RecEmpty(203):レシート用紙（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Blackmark(10001):ブラックマークエラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Fatal (10003):フェイタルエラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Overheat (10006):ヘッド昇温エラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。）
その他	**ErrorCode 一覧を参照してください。**

EndInsertion メソッド

形式

void EndInsertion ();

説明

このメソッドはスリッププリンターを対象としているため、本クラスライブラリではサポートされません。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
Illegal(106)	POS プリンターはスリップを持っていません。
その他	**ErrorCode 一覧を参照してください。**

EndRemoval メソッド

形式

void EndRemoval ();

説明

このメソッドはスリッププリンターを対象としているため、本クラスライブラリではサポートされません。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
Illegal(106)	POS プリンターはスリップを持っていません。
その他	**ErrorCode 一覧を参照してください。**

MarkFeed メソッド

形式

void MarkFeed (PrinterMarkFeeds *Type*);

Type パラメータは、マーク付き用紙のハンドリングの種類を示します。

説明

このメソッドはサポートされていません。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
Illegal(106)	レシート印刷ステーションは、マーク付き用紙のハンドリング機能をサポートしていません。（**CapRecMarkFeed** プロパティを参照）
その他	**ErrorCode 一覧を参照してください。**

PageModePrint メソッド

形式

void PageModePrint (PageModePrintControl *Control*);

パラメータ	説明
Control	ページモード指定。下記の値を参照。

Control パラメータの値は以下の通りです。

値	意味
PageMode(1)	ページモード開始
Normal(3)	ページモード印刷領域の印刷データを印刷し、その印刷データを消去し、ページモードを終了します。
Cancel(4)	ページモード印刷領域の印刷データを消去し、何も印刷せずにページモードを終了します。

説明

PageModeStation プロパティに指定したステーションを対象として、ページモードを開始／終了します。

Control に PageMode(1) を指定した場合、ページモードを開始します。その後、Normal(3)、Cancel(4)を指定して **PageModePrint** メソッドが呼び出されるまでの間、**PrintNormal**、**PrintBarCode**、**PrintBitmap**、**PrintMemoryBitmap** メソッドで印刷データをバッファリングすることができます。（この間に呼び出されたメソッドは、印刷データをバッファリングするだけで、印刷を開始することはありません。また、**AsyncMode** プロパティの設定は、ページモード機能に影響を与えません。それぞれの操作に対して、**OutputID** が割り当てられることも、**OutputCompleteEvent** が通知されることもありません。）

Control に Normal(3)を指定した場合、ページモードを終了し通常の状態へ遷移します。**PrintNormal**、**PrintBarCode**、**PrintBitmap**、**PrintMemoryBitmap** メソッドでバッファリングした印刷データがあれば、それを印刷します。バッファリングした印刷データは保存されません。

Control に PTR_PM_CANCEL を指定した場合、ページモードを終了し、通常の状態へ遷移します。**PrintNormal**、**PrintBarCode**、**PrintBitmap**、**PrintMemoryBitmap** メソッドでバッファリングした印刷データがあっても、それらは印刷されず、保存もされません。

PageModePrint メソッドが呼び出された時、**PageModePrintArea** で定義されたページモード印刷領域上のすべての印刷データが印刷され、ページモード印刷領域の下端まで紙送りされます。
複数のページモード印刷領域が定義されていた場合は、それぞれのページモード印刷領域上のすべての印刷データが印刷され、最も下方に配置されているページモード印刷領域の下端まで紙送りされます。

ページモード領域全体が、1 つのトランザクションとして一括処理されます。
AsyncMode プロパティが TRUE であれば非同期的に、FALSE であれば同期的に処理されます。

ClearOutput メソッドが呼び出されると、ページモードを終了し通常の状態へ遷移します。このとき、バッファリングした印刷データも消去されます。

ページモード機能は、トランザクション印刷と回転印刷を同時に利用することはできません。（詳細については、「10. ページモード機能について『PageModePrint メソッドの排他』」参照）
また、ページモード開始中、エスケープシーケンスによる文字位置（左寄せ、中央揃え、右寄せ）の指定は無効となります。水平方向の印刷位置の調整は PageModeHorizontalPosition プロパティ指定により行います。
ページモード開始中に再度ページモードを開始した場合、それまでバッファリングした印刷データは破棄されます。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
例外なし	メソッドは正常終了しました。
Illegal(106)	以下のエラーの一つが起こりました。 ・*Control* が存在しません。 ・*Contorol* が、不正な値です。 ・トランザクション印刷開始中です。 ・回転印刷開始中です
NoHardware(107)	POS プリンターはオフ、またはオフラインです。
Failure(111)	本クラスライブラリはエラー状態です。エラー状態を解除して実行してください。
Busy(113)	出力中なので実行できません。
Extended(114)	**ErrorCodeExtended** = CoverOpen(201): POS プリンターカバーが開いています。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = RecEmpty(203):レシート用紙がありません。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Blackmark(10001):ブラックマークエラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Fatal (10003):フェイタルエラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** =Overheat (10006):ヘッド昇温エラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。）
その他	**ErrorCode** の項目を参照してください

PrintBarCode メソッド

形式

void PrintBarCode (PrinterStation *Station*, string *Data*, BarCodeSymbology *Symbology*, int *Height*,
int *Width*, int *Alignment*, BarCodeTextPosition *TextPosition*);

パラメータ	説明
Station	Receipt(2)を指定
Data	バーコードの文字列。
Symbology	使用されるバーコードタイプ。（以下の値を参照）
Height	バーコードの高さ。**MapMode** で定義された単位で表記します。1～255dot まで設定可能です。 PDF417 印刷時は、正立、倒立時は 12～831dot、左右 90 度回転時には 12～**RecLineWidth** dot の範囲で設定可能です。 2 次元バーコードの QR コード、マイクロ QR コードについては、モジュール幅として 1～16 の範囲を指定してください。それ以外の値をした場合、Illegal(106)が通知されます。
Width	バーコードの幅。**MapMode** で定義された単位で表記します。正立／倒立の場合は、**RecLineWidth** プロパティの値まで設定可能です。
Alignment	バーコードの位置。下記の値を参照。
TextPosition	文字列の位置。以下の値を参照。

本リリースでの *Symbology* パラメータの値は以下の通りです。

値	ラベルタイプ
Upca(101)	UPC-A
Upce(102)	UPC-E
Ean8(103)	EAN 8 (= JAN 8)
Jan8(103)	JAN 8 (= EAN 8)
Ean13(104)	EAN 13 (= JAN 13)
Jan13(104)	JAN 13 (= EAN 13)
Itf(106)	Interleaved 2 of 5
Codabar(107)	Codabar(NW-7)
Code39(108)	Code 39
Code93(109)	Code 93
Code128(110)	Code 128
Pdf417(201)	PDF417
QRCode(204)	QR code (2 次元バーコード)
MicroQRCode(205)	Micro QR code (2 次元バーコード)

Alignment パラメータの値は、以下の通りです。

値	意味
Left(-1)	左寄せ（印字データに対して実行するので、180 度回転印字時は、POS プリンター印字方向に対しては右寄せとなります。）
Center(-2)	センタリング 2 次元バーコードの場合、左右 90 度回転中はこのパラメータをサポートしません。Left(-1)として動作します。
Right(-3)	右寄せ（印字データに対して実行するので、180 度回転印字時は、POS プリンター印字方向に対しては左寄せとなります。） 2 次元バーコードの場合、左右 90 度回転中はこのパラメータをサポートしません。Left(-1)として動作します。
その他	バーコード印刷を開始する左端からの距離。**MapMode** で定義された単位で表記します。*Width* パラメータで指定されたバーコード幅から本クラスライブラリが算出する実際のバーコード印字幅+左端からの距離が **RecLineWidth** プロパティの値を超える場合、Illegal(106)が返ります。但し、**RotateSpecial** プロパティで、Right90(257), Left90(258)が指定されていた場合、Left(-1)が指定されたとみなして印字を行います。 PDF417 の場合、左右 90 度回転中はこのパラメータをサポートしません。Left(-1)として動作します。

TextPosition パラメータの値は、以下の通りです。

値	意味
None(-11)	テキストは印字しない。バーコードのみ印刷します。
Above(-12)	バーコードの上にテキストを印字します。
Below(-13)	バーコードの下にテキストを印字します。

説明

指定した POS プリンターにバーコードを印刷する時にこのメソッドを呼び出します。

このメソッドは、**AsyncMode** が **FALSE** ならば、同期的に実行され、 **AsyncMode** が **TRUE** ならば、非同期的に実行されます。

ページモード開始中に PrintBarCode メソッドでビットマップ印刷時、Alignment 指定は以下の動作となります。

値	意味
Left(-1)	左寄せ
Center(-2)	センタリング
Right(-3)	右寄せ
その他	ビットマップ印刷を開始する左端からの距離。**MapMode** で定義された単位で表記します。*Width* との合計値が、*Width* パラメータの制限を超えてはいけません。

Symbology 毎のバーコード印字可能条件を以下に示します。回転印字時には、2 色印字可能かどうか(**CapRec2Color**=TRUE)によって、設定できる値が異なります。

Symbology	印字可能な キャラクタ種別	正立／倒立モード時		左右 90 度 回転モード時	
		文字列長	Width(dots)	文字列長	Width(dots)
Upca	‘0’-‘9’ の 10 種	11～ 12	95～ RecLineWidth 値	11～ 12	95～1662
Upce		11～ 12	51～ RecLineWidth 値	11～ 12	51～1662
Jan8		7～8	67～ RecLineWidth 値	7～8	67～1662
Jan13		12～ 13	95～ RecLineWidth 値	12～ 13	95～1662
Code39	‘0’～’9’, ‘A’～’Z’, 空白, ‘$’, ‘%’, ‘+’, ‘-’, ‘.’, ‘/’ の 43 種 （スタート・ストップキャラクタ’*’は自動で付加されます。）	1～34	47～ RecLineWidth 値	1～ 101	47～1662
Itf	‘0’-‘9’ の 10 種	2～62	27～ RecLineWidth 値	2～ 182	27～1662
Codaber	0’～’9’, ‘A’～’D, ‘$’, ‘+’, ‘-’, ‘.’, ‘/’, ‘:’ の 20 種	3～47	41～ RecLineWidth 値	3～ 138	41～1662
Code93	0x00～0x7F の 128 種 （下段は 2 キャラクタ時）	1～59	46～ RecLineWidth 値	1～ 88	46～1662
		1～29		1～ 44	
Code128	Code Set A: 0x00 ～ 0x5F Code Set B 0x20 ～ 0x7F Code Set C 0x00 ～ 0x63 但し、”{“を含むキャラクタは除く。詳細は後述。	3～51	46～ RecLineWidth 値	3～ 74	46～1662

Symbology	印字可能な キャラクタ種別	正立／倒立モード時		左右 90 度 回転モード時	
		文字列長	Width(dots)	文字列長	Width(dots)
Pdf417	0x00～0xFF までの 256 種。ただし、0x00～0x7F までは ASCII コードに従い、0x80～0xFF までは PC437(USA:Standard Europe)の英語表の拡張キャラクタセットに従う文字列。	1 ～ 1069	172 ～ RecLineWidth 値	1 ～ 1069	172～831
QRCode	数字(‘0’～‘9’)、 大文字(‘A’～’Z’)、特殊記号(スペース、‘$’, ‘%’, ‘*’ ,‘+’, ‘-’, ‘.’, ‘/’, ‘:’)、バイナリ(0x00～0xFF)、漢 字 (0x8140 ～ 0x9FFC 、 0xE040 ～ 0xEBBF)	1 ～ 7089	21 ～ RecLineWidth 値	1 ～ 7089	21 ～ RecLineWidth 値
MicroQRCode	数字(‘0’～‘9’)、 大文字(‘A’～’Z’)、特殊記号(スペース、‘$’, ‘%’, ‘*’ ,‘+’, ‘-’, ‘.’, ‘/’, ‘:’)、バイナリ(0x00～0xFF)、漢 字 (0x8140 ～ 0x9FFC 、 0xE040 ～ 0xEBBF)	1～35	11 ～ RecLineWidth 値	1 ～ 35	11 ～ RecLineWidth 値

各バーコードの印字幅決定アルゴリズムを以下に示します。最終的なバーコードの印字幅(dot)は、パラメータを変更していき、**PrintBarcode** の *Width* パラメータに指定された値を超えない範囲でもっとも近い値で印字を行います。

Symbology	印字幅計算式
Upca	バーコード幅 = 95 * dotNarrow
Upce	バーコード幅 = 51 * dotNarrow
Jan8	バーコード幅 = 67 * dotNarrow
Jan13	バーコード幅 = 95 * dotNarrow
Code39	バーコード幅 = 6 * dotNarrow + 3 * dotWide + 1 * dotNarrow + (6 * dotNarrow + 3 * dotWide + 1 dotNarrow)* Length + 6 * dotNarrow + 3 * dotWide （Length = 印字キャラクタ数）
Itf	バーコード幅 = 4 * dotNarrow + (3 * dotNarrow + 2 * dotWide) * Length + 2 * dotNarrow + 1 * dotWide （Length = 印字キャラクタ数）
Codaber	バーコード幅 = (5 * dotNarrow + 2 * dotWide) * (Length – Wlen) + (4 * dotNarrow + 3 * dotWide) * Wlen + 1 * dotNarrow * (Length + 1) （Length = 印字キャラクタ数） （Wlen = “:”, “/”, “.”, “+”, “A”,”B”,”C”,”D” の文字数）
Code93	バーコード幅 = 9 * dotNarrow + 9 * dotNarrow * Wlen + (9 * dotNarrow) * 2 * (Length - Wlen) + 9 * 2 * dotNarrow + 10 * dotNarrow （Length = 印字キャラクタ数） （Wlen = “0”～”9”,”A”～”Z”,” “,”$”,”%”,”+”,”-“～”/”の文字数）
Code128	バーコード幅 = 11 * dotNarrow * (Length + 1) + 13 * dotNarrow （Length =印字キャラクタ数 - 特殊キャラクタ※） ※ “{“ が先頭についているキャラクタ数。”{AA{BA”が指定されたときは、Length = 6 – 2 = 4 となります。

Symbology	印字幅計算式
Pdf417	バーコード幅 = ((17 * (C + 2)) + (17 + 18)) * X バーコードの高さ = RYX C：カラム数 X：公称細エレメント幅 R：段数 Y：段の高さ ※Width 及び Height パラメータの設定より、段数とカラム数を決定してWidth パラメータを超えない最大サイズでコードを生成します。 但し、Width パラメータの設定を最優先してカラム数を算出しているためHeight パラメータは設定した高さと一致しない場合があります。
QRCode	Bar code width = 縦セル数 * ﾓｼﾞｭｰﾙ幅(1～16) Bar code height = 横セル数 * ﾓｼﾞｭｰﾙ幅(1～16) ※バーコード幅、高さは、Width の設定値を超えない最大サイズになります。
MicroQRCode	Bar code width = 縦セル数 * ﾓｼﾞｭｰﾙ幅(1～16) Bar code height = 横セル数 * ﾓｼﾞｭｰﾙ幅(1～16) ※バーコード幅、高さは、Width の設定値を超えない最大サイズになります。

※dotNarrow と dotWide の関係

dotNarrow	1	2	3	4	5	6
dotWide	3	5	9	11	14	18

バーコード印字注意点について

1. CODE39 を印字する場合、スタートキャラクタ、ストップキャラクタの”*”は自動的に付加されますので、キャラクタに設定する必要はありません。

2. ITF を指定する場合、キャラクタ数は偶数で指定する必要があります。奇数を指定した場合、Illegal(106)が返ります。

3. CODABER を指定する場合、キャラクタの先頭と末尾はかならず、”A”～”D”のいずれかである必要があります。よって、その間に任意のキャラクタを設定した合計 3 文字以上で指定する必要があります。これに従っていない場合は、Illegal(106)が返ります。

4. UPC-E を指定する場合、展開は下記の表に基づいて行われます。UPC-A レフトコードは，先頭 2～6 キャラクタ、UPC-A ライトコードは、 先頭 7～11 キャラクタ目を示し、短縮されるコードが実際に UPC-E として印字されます。UPC-A 先頭キャラクタが 0 以外か、下記の表に基づかないキャラクタが指定された場合、Illegal(106)が返ります。

   例　05810000226 → 58226 に変換されます。
   　　09859363583 → Illegal(106)が返ります。

メーカーコード UPC-A のレフトコード					アイテムコード UPC-A のライトコード					短縮コード					
F1	F2	F3	F4	F5	A1	A2	A3	A4	A5	Z1	Z2	Z3	Z4	Z5	Z6
0-9	0-9	0	0	0	0	0	0-9	0-9	0-9	F1	F2	A3	A4	A5	0
0-9	0-9	1	0	0	0	0	0-9	0-9	0-9	F1	F2	A3	A4	A5	1
0-9	0-9	2	0	0	0	0	0-9	0-9	0-9	F1	F2	A3	A4	A5	2
0-9	0-9	3-9	0	0	0	0	0	0-9	0-9	F1	F2	F3	A4	A5	3
0-9	0-9	0-9	1-9	0	0	0	0	0	0-9	F1	F2	F3	F4	A5	4
0-9	0-9	0-9	0-9	1-9	0	0	0	0	5-9	F1	F2	F3	F4	F5	A5

5. CODE128 を印字する場合は以下のようにキャラクタを設定します。

   1. バーコードの先頭は，"{A", "{B", "{C" のいずれかを指定してください。以下をそれぞれ，CODE A, CODE B, CODE C に設定します。
   2. ファンクションコードを指定するには、"{1", "{2", "{3", "{4" を指定します。それぞれ、FNC1，FNC2, FNC3 FNC4 の指定となります。尚、CODE C では、FUNC1 のみ有効です。CODE C で、FUNC1 以外を指定すると Illegal(106)が返ります。
   3. CODE B で、"{" を印字するには，"{{" を指定してください。
   4. SHIFT を設定するには、"{S" を指定してください。以後の 1 キャラクタのコードセットが、CODE A ←→ CODE B のようにシフトします。CODE C で指定した場合、Illegal(106)が返ります。

CODE A,CODE B,CODE C で印字可能なキャラクタを次に示します。

印字するキャラクタ			印字するキャラクタ		
CODE-A	CODE-B	CODE-C	CODE-A	CODE-B	CODE-C
SPACE	SPACE	00(00H)	U	U	53(35H)
!	!	01(01H)	V	V	54(36H)
“	“	02(02H)	W	W	55(37H)
#	#	03(03H)	X	X	56(38H)
$	$	04(04H)	Y	Y	57(39H)
%	%	05(05H)	Z	Z	58(3AH)
&	&	06(06H)	[	[	59(3BH)
‘	‘	07(07H)	/	/	60(3CH)
(	(	08(08H)	]	]	61(3DH)
)	)	09(09H)	^	^	62(3EH)
*	*	10(0AH)	_	_	63(3FH)
+	+	11(0BH)	NULL(00H)	`	64(40H)
,	,	12(0CH)	SOH(01H)	a	65(41H)
-	-	13(0DH)	STX(02H)	b	66(42H)
.	.	14(0EH)	ETX(03H)	c	67(43H)
/	/	15(0FH)	EOT(04H)	d	68(44H)
0	0	16(10H)	ENG(05H)	e	69(45H)
1	1	17(11H)	ACK(06H)	f	70(46H)
2	2	18(12H)	BEL(07H)	g	71(47H)
3	3	19(13H)	BS(08H)	h	72(48H)
4	4	20(14H)	HT(09H)	i	73(49H)
5	5	21(15H)	LF(0AH)	j	74(4AH)
6	6	22(16H)	VT(0BH)	k	75(4BH)
7	7	23(17H)	FF(0CH)	l	76(4CH)
8	8	24(18H)	CR(0DH)	m	77(4DH)
9	9	25(19H)	SO(0EH)	n	78(4EH)
:	:	26(1AH)	SI(0FH)	o	79(4FH)
;	;	27(1BH)	DLE(10H)	p	80(50H)
<	<	28(1CH)	DC1(11H)	q	81(51H)
=	=	29(1DH)	DC2(12H)	r	82(52H)

印字するキャラクタ			印字するキャラクタ		
CODE-A	CODE-B	CODE-C	CODE-A	CODE-B	CODE-C
>	>	30(1EH)	DC3(13H)	s	83(53H)
?	?	31(1FH)	DC4(14H)	t	84(54H)
@	@	32(20H)	NAK(15H)	u	85(55H)
A	A	33(21H)	SYN(16H)	v	86(56H)
B	B	34(22H)	ETB(17H)	w	87(57H)
C	C	35(23H)	CAN(18H)	x	88(58H)
D	D	36(24H)	EM(19H)	y	89(59H)
E	E	37(25H)	SUB(1AH)	z	90(5AH)
F	F	38(26H)	ESC(1BH)	{ “{{“	91(5BH)
G	G	39(27H)	FS(1CH)	|	92(5CH)
H	H	40(28H)	GS(1DH)	}	93(5DH)
I	I	41(29H)	RS(1EH)	~	94(5EH)
J	J	42(2AH)	US(1FH)	DEL	95(5FH)
K	K	43(2BH)			96(60H)
L	L	44(2CH)			97(61H)
M	M	45(2DH)			98(62H)
N	N	46(2EH)			99(63H)
O	O	47(2FH)	以下は、”{“を指定して利用します。		
P	P	48(30H)	FNC 3 “{3”	FNC 3 “{3”	
Q	Q	49(31H)	FNC 2 “{2”	FNC 2 “{2”	
R	R	50(32H)	SHIFT “{S”	SHIFT “{S”	
S	S	51(33H)	CODE C “{C”	CODE C “{C”	
T	T	52(34H)	CODE B “{B”	CODE A “{A”	CODE B “{B”
			FNC 4 “{4”	FNC 4 “{4”	CODE A “{A”
			FNC 1 “{1”	FNC 1 “{1”	FNC 1 “{1”

6. *TextPosition* の指定と, *Width* によるバーコード印字可否条件を下記に示します。*Width* パラメータについて, 特に記述のないものは, 印字可能範囲内であれば *TextPosition* によらずバーコードが印字可能ということを示しています。下記の表は, *Width* パラメータが dot 単位で且つ、印字可能範囲であることが前提です。

Symbology	***TextPosition*** **None**	***TextPosition*** **Above** **Below**
Upca(101)	印字可	Width=95～189 ILLEGAL
Upce(102)	印字可	Width=51～101 ILLEGAL
Jan8(103)	印字可	Width=67～133 ILLEGAL
Jan13(104)	印字可	Width=95～189 ILLEGAL
Code39(108)	印字可	印字可
Itf(106)	印字可	印字可
Codabar(107)	印字可	印字可
Code93(109)	※1	※1
Code128(110)	※2	※2

※1 ... Illegal(106)となる, *width* と キャラクタ数の関係は

37 + 9 * wlen + 18(len - wlen) <= *width* < 74 + 18 * wlen + 36(len - wlen)

の範囲内です。

・wlen = キャラクタ内に含まれる '0' - '9', 'A' - 'Z', ' ', '$', '%', '+', '-', '.', '/' の数

・len = キャラクタ長

※2 ... Illegal(106)となる, width と キャラクタ数の関係は

24 + 11 * len <= width < 48 + 22 * len

の範囲内です.

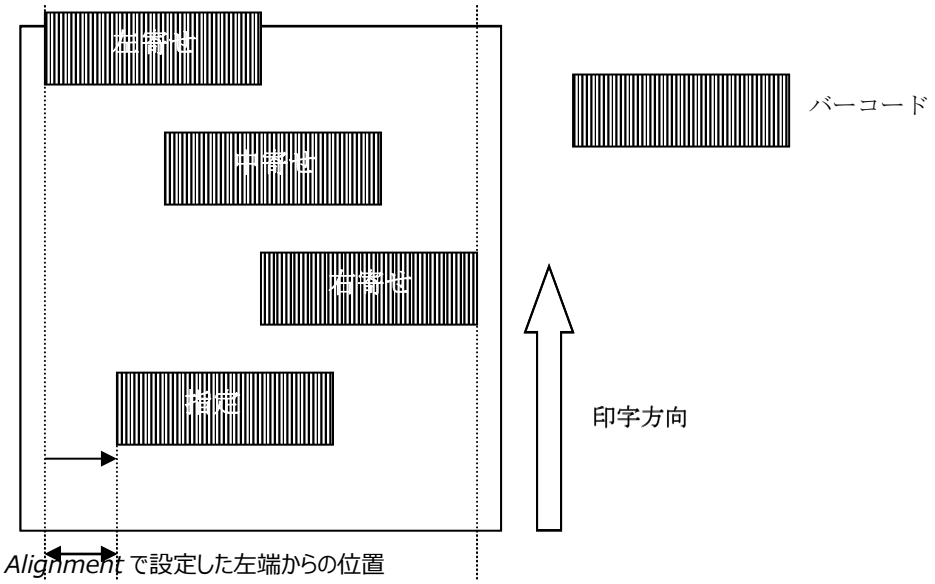
・len = キャラクタ長 から、それに含まれる

"{A", "{B", "{C", "{1", "{2", "{3", "{4", "{S", "{{" の数を引いた値

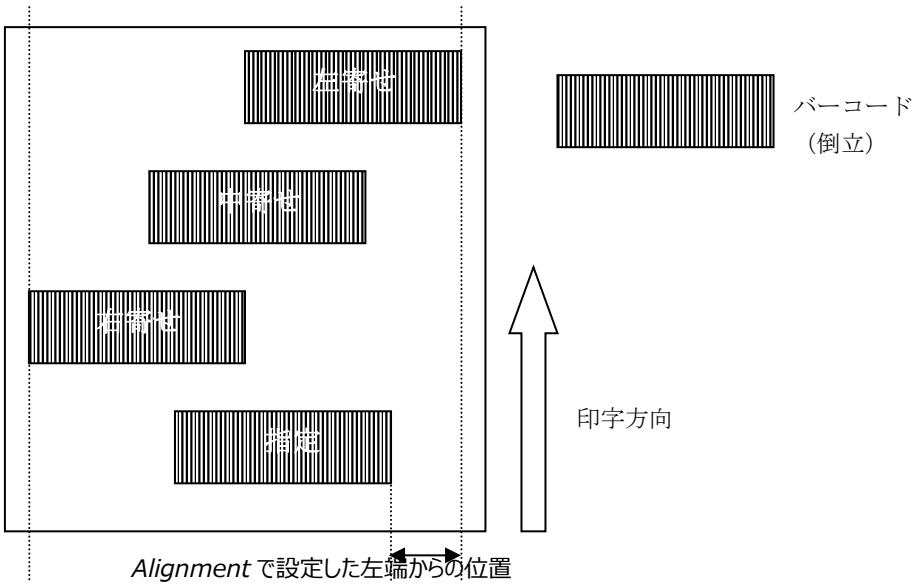
7. PDF417 において、DirectIO Command= OPOS_FIT_DIO_SET_PDFVHSIZE(26)で縦横サイズを指定した場合、段数（縦サイズ）×コードワード桁数（横サイズ）が 928 を超える場合、バーコードの印刷はされず、Illegal(106)が返ります。

RotateSpecial を使用したバーコードの回転印字について

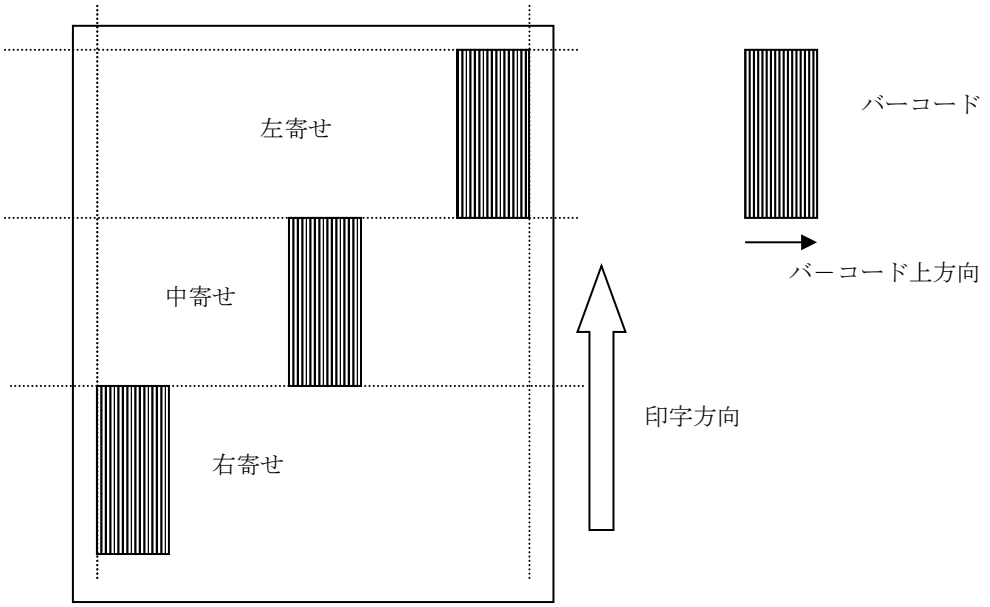
正立印字時の、*Alignment* 指定による印字位置の変化



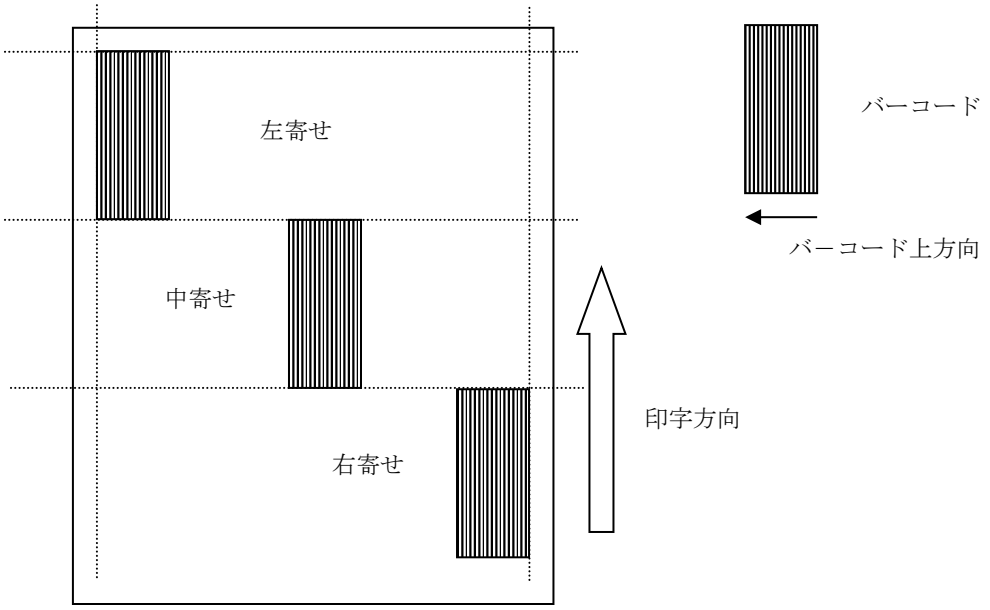
倒立印字時の、*Alignment* 指定による印字位置の変化



右９０度回転時の、*Alignment* 指定による印字位置の変化



左９０度回転時の、*Alignment* 指定による印字位置の変化



※二次元バーコードについては、左右 90 度回転時の *Aligment* はすべて Left(-1) 固定となります。

値	意味
例外なし	メソッドは正常終了しました。
Illegal(106)	以下のエラーの一つが起こりました。 ・*Station* が存在しません。 ・*Station* がバーコード印刷をサポートしていません。 ・*Height* あるいは *Width* が、0 あるいは大きすぎます。 ・サポートされていない *Symbology* です。 ・*Symbology* でサポートされていない文字があります。 ・*Alignment* が、不正な値であるか、大きすぎます（*Alignment* を絶対位置指定した場合は、*Alignment* の指定値と、バーコードの実際の印字幅（*Width* に一番近い値で計算した値）の合計値が、印字可能幅を超える場合）。 ・*TextPosition* が、不正な値です。
NoHardware(107)	POS プリンターはオフ、またはオフラインです。
Failure(111)	本クラスライブラリはエラー状態です。エラー状態を解除して実行してください。
Busy(113)	出力中なので実行できません。
Extended(114)	**ErrorCodeExtended** = CoverOpen(201): POS プリンターカバーが開いています。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = RecEmpty(203):レシート用紙がありません。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Blackmark(10001):ブラックマークエラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Fatal (10003):フェイタルエラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Overheat (10006):ヘッド昇温エラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。）
その他	**ErrorCode 一覧**の項目を参照してください

PrintBitmap メソッド

形式

void PrintBitmap (PrinterStation *Station*, string *FileName*, int *Width*, int *Alignment*);

パラメータ	説明
Station	Receipt(2)を指定
FileName	Windows ビットマップファイル名。ファイルは圧縮されたフォーマットではいけません。（フルパスまたは相対パス指定）
Width	ビットマップの印刷幅。下記の値を参照。
Alignment	ビットマップの印刷位置。下記の値を参照。

Width パラメータの値は、以下の通りです。

値	意味
Asis(-11)	POS プリンターのドット当たり 1 ビットマップピクセルで、ビットマップを印刷します。
その他	ビットマップ幅。**MapMode** で定義された単位で表記します。有効値は、1～RecLineWidth プロパティの値までです。

Alignment パラメータの値は、以下の通りです。

値	意味
Left(-1)	左寄せ
Center(-2)	センタリング
Right(-3)	右寄せ
その他	ビットマップ印刷を開始する左端からの距離。**MapMode** で定義された単位で表記します。*Width* との合計値が、*Width* パラメータの制限を超えてはいけません。

説明

指定されたプリンターにビットマップを印刷する時にこのメソッドを呼び出します。読み込んだビットマップを、モノクロームまたは 2 色にして印刷します。

２色印刷設定の場合、黒を第一色、その他を第二色で印刷します。

登録可能なビットマップは、横サイズ(*Width*)が **RecLineWidth** dot 以下（*Alignment* が絶対位置指定の場合、*Width*+ *Alignment*<= **RecLineWidth**）、且つ縦サイズが単色データは 1662dot、2 色設定で 2 色データの場合は 831dot 以下となります。

PrintBitmap は呼び出された時点でビットマップデータをプリンターに送信するため、良好なパフォーマンスが得られません。**SetBitmap** とエスケープシーケンスによるビットマップの印刷を推奨します。

このメソッドは、**AsyncMode** が **FALSE** ならば、同期的に実行され、**AsyncMode** が **TRUE** ならば、非同期的に実行されます。

Width パラメータはビットマップの変換をコントロールします。もし、*Width* が Asis ならば変換は実行されません。ビットマップは POS プリンタードット当たり 1 ビットマップピクセルで印刷されます。

もし *Width* が 0 でなければ、ビットマップは、その幅が指定された幅でそのアスペクト比が変化しないように、拡大されたり圧縮されたりすることにより、変換されます。

※指定されたビットマップデータがモノクロの場合、モノクロビットマップがプリンターに設定されます。モノクロ以外のデータの場合、**CapRec2Color** プロパティが TRUE の場合は 2 色ビットマップ印字が実行され、FALSE の場合はモノクロデータとして印字されます。

ページモード開始中に PrintBitmap メソッドでビットマップ印刷時、Alignment 指定は以下の動作となります。

値	意味
Left(-1)	左寄せ
Center(-2)	センタリング
Right(-3)	右寄せ
その他	ビットマップ印刷を開始する左端からの距離。**MapMode** で定義された単位で表記します。*Width* との合計値が、*Width* パラメータの制限を超えてはいけません。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
例外なし	メソッドは正常終了しました。
Illegal(106)	以下のエラーの内、一つが起こりました。 ・*Station* が存在しません。 ・*Station* がビットマップ印刷をサポートしていません。 ・*Width* が大きすぎます。 ・*Alignment* が、不正な値であるか、大きすぎます。
NoHardware(107)	POS プリンターはオフ、またはオフラインです。
Failure(111)	本クラスライブラリはエラー状態です。エラー状態を解除して実行してください。
Noexist(109)	*FileName* で指定されたファイルが見つかりませんでした。
Busy(113)	出力中なので実行できません。（**AsyncMode** が **FALSE** の時のみ返されます。）
Extended(114)	**ErrorCodeExtended** = CoverOpen(201): POS プリンターカバーが開いています。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = RecEmpty (203): 指定されたレシートは、用紙がありません。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Toobig (206): 指定したビットマップが大きすぎます。 印字可能なビットマップサイズは、横 **RecLineWidth** プロパティ dot、縦 モノクロ時 1662 dot ２色設定時 831 dot の間 です。 **ErrorCodeExtended** = Badformat (207): 指定したビットマップのフォーマットが異なっています。指定さ れたファイルはビットマップファイルではありません。 ２色印字指定でデータが２４ビットビットマップの場合。 （モノクロ指定で２４ビットビットマップは印刷可能） **ErrorCodeExtended** = Blackmark(10001):ブラックマークエラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Fatal (10003):フェイタルエラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Overheat (10006):ヘッド昇温エラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。）
その他	**ErrorCode 一覧を参照してください。**

PrintImmediate メソッド

形式

void PrintImmediate (PrinterStation *Station*, string *Data*);

パラメータ	説明
Station	Receipt(2)を指定
Data	印刷される文字。プリント可能な文字とエスケープシーケンス、キャリッジリターン(13 decimal)、ニューライン/ラインフィード(10 decimal)から成ります。

説明

POS プリンターに *Data* を印刷する時にこのメソッドを呼び出します。非同期印字実行中(**State**=Busy(3)中)はBusy(113)が返ります。エラーイベント中(**State**=Error(4)中) は Failure(111)が返ります。また、最大桁数を越える分の印刷データは次の行に折り返して印刷します。

Data 内の特別なキャラクタの値は、以下の通りです。

値	意味
Newline/Line Feed (10)	バッファ内のデータを印字後、次の行へ進みます。(キャリッジリターンは、その行を印刷するためには必要ありません。)
Carriage Return(13)	もし、キャリッジリターンがラインフィードのすぐ前にあれば、キャリッジリターンは無視されます。 キャリッジリターンはラインフィードのように振る舞います。 **ValidateData** メソッドは、ラインフィード無しのキャリッジリターンが可能か否かと、そのために逆フィードがサポートされる必要があるか否かを決定するために利用されます。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
例外なし	メソッドは正常終了しました。
Illegal(106)	指定した POS プリンター（レシート以外）が存在しません。
NoHardware(107)	POS プリンターはオフ、またはオフラインです。
Failure(111)	本クラスライブラリはエラー状態です。エラー状態を解除して実行してください。
Busy(113)	出力中なので実行できません。
Extended(114)	**ErrorCodeExtended** = CoverOpen(201): POS プリンターカバーが開いています。 **ErrorCodeExtended** = Recempty(203):レシート用紙がありません。 **ErrorCodeExtended** = Blackmark(10001):ブラックマークエラーが発生しました。 **ErrorCodeExtended** = Fatal (10003):フェイタルエラーが発生しました。 **ErrorCodeExtended** = Overheat (10006):ヘッド昇温エラーが発生しました。
その他	**ErrorCode 一覧**の項目を参照してください

PrintMemoryBitmap メソッド

形式

void PrintMemoryBitmap (PrinterStation *Station*, System.Drawing.Bitmap *Data*,–int *Width*,
int Alignment);

パラメータ	説明
Station	Receipt(2)を指定
Data	ビットマップデータを保持しているバイト配列。
Width	ビットマップの印刷幅。下記の値を参照。
Alignment	ビットマップの印刷位置。下記の値を参照。

Width パラメータの値は、

値	意味
Asis(-11)	POS プリンターのドット当たり 1 ビットマップピクセルで、ビットマップを印刷します。
その他	ビットマップ幅。MapMode で定義された単位で表記します。

Alignment パラメータの値は、

値	意味
Left(-1)	左寄せ。
Center(-2)	センタリング。
Right(-3)	右寄せ。
その他	ビットマップ印刷を開始する左端からの距離。 MapMode で定義された単位で表記します。

説明

指定されたステーションに対して、メモリ上に保持されたビットマップを印刷する時にこのメソッドを呼び出します。バイト配列へのポインタとして渡されたビットマップを、モノクロームまたは 2 色にして印刷します。

2 色印刷設定の場合、黒を第一色、その他を第二色で印刷します。

印刷可能なビットマップは、横サイズ(*Width*)が **RecLineWidth** dot 以下（*Alignment* が絶対位置指定の場合、*Width*+ *Alignment*<= **RecLineWidth**）、且つ縦サイズが単色データは 1662dot、2 色設定で 2 色データの場合は 831dot 以下となります。

このメソッドは、**AsyncMode** が **FALSE** ならば、同期的に実行され、 **AsyncMode** が **TRUE** ならば、非同期的に実行されます。

Width パラメータはビットマップの変換をコントロールします。もし、*Width* が Asis ならば、変換は実行されません。ビットマップは POS プリンタードット当たり 1 ビットマップピクセルで印刷されます。このオプションの長所は、以下のことです。

高いパフォーマンスのビットマップ印刷ができます。

ビットマップは、水平方向と垂直方向のドット単位がプリンター固有のアスペクト比に調整され、良好な印刷が得られます。

もし *Width* が 0 でなければ、ビットマップは、その幅が指定された幅でそのアスペクト比が変化しないように、拡大されたり圧縮されたりすることにより、変換されます。

また、TransactionPrint にはバッファリングされないため、バッファリング中においても、データをプリンターに送信します。

印刷可能な画像形式は PrintBitmap プロパティを参照して下さい。

ページモード開始中の PrintMemoryBitmap メソッドはサポートしません。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
例外なし	メソッドは正常終了しました。
Illegal(106)	以下のエラーの内、一つが起こりました。 ・*Station* が存在しません。 ・*Station* がビットマップ印刷をサポートしていません。 ・*Width* が大きすぎます。 ・*Alignment* が、不正な値であるか、大きすぎます。
NoHardware(107)	POS プリンターはオフ、またはオフラインです。
Failure(111)	本クラスライブラリはエラー状態です。エラー状態を解除して実行してください。
Busy(113)	出力中なので実行できません。（**AsyncMode** が **FALSE** の時のみ返されます。）
Extended(114)	**ErrorCodeExtended** = CoverOpen(201): POS プリンターカバーが開いています。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Recempty(203): 指定されたレシートは、用紙がありません。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Toobig (206): 指定したビットマップが大きすぎます。 印字可能なビットマップサイズは、横 **RecLineWidth** プロパティ dot、縦 モノクロ時 1662 dot 2 色設定時 831 dot の間 です。 **ErrorCodeExtended** = Badformat (207): 指定したビットマップのフォーマットが異なっています。指定さ れたファイルはビットマップファイルではありません。 **ErrorCodeExtended** =Blackmark(10001):ブラックマークエラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Fatal (10003):フェイタルエラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Overheat (10006):ヘッド昇温エラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。）
その他	**ErrorCode 一覧を参照してください。**

PrintNormal メソッド

形式

void PrintNormal (PrinterStation *Station*, string *Data*);

パラメータ	説明
Station	Receipt(2)を指定
Data	印刷される文字。プリント可能な文字とエスケープシーケンス、キャリッジリターン(13 decimal)、ニューライン/ラインフィード(10 decimal)から成ります。

説明

レシートプリンターに *Data* を印刷する時にこのメソッドを呼び出します。最大桁数を越える分の印刷データは次の行に折り返して印刷します。

このメソッドは、**AsyncMode** が **FALSE** ならば、同期的に実行され、**AsyncMode** が **TRUE** ならば、非同期的に実行されます。

Data 内の特別なキャラクタの値は、以下の通りです。

値	意味
Newline/Line Feed (10)	バッファ内のデータを印字後、次の行へ進みます。(キャリッジリターンは、その行を印刷するためには必要ありません。)
Carriage Return(13)	もし、キャリッジリターンがラインフィードのすぐ前にあれば、キャリッジリターンは無視されます。 キャリッジリターンはラインフィードのように振る舞います。 **ValidateData** メソッドは、ラインフィード無しのキャリッジリターンが可能か否かと、そのために逆フィードがサポートされる必要があるか否かを決定するために利用されます。

例外

このメソッドを呼び出したときに、次の PosControlException をスローすることがあります。

値	意味
例外なし	メソッドは正常終了しました。
Illegal(106)	指定した POS プリンター（レシート以外）が存在しません。
NoHardware(107)	POS プリンターはオフ、またはオフラインです。
Failure(111)	本クラスライブラリはエラー状態です。エラー状態を解除して実行してください。
Busy(113)	出力中なので実行できません。
Extended(114)	**ErrorCodeExtended** = CoverOpen(201): POS プリンターカバーが開いています。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Recempty(203):レシート用紙がありません。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Blackmark(10001):ブラックマークエラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Fatal (10003):フェイタルエラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Overheat (10006):ヘッド昇温エラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。）
その他	**ErrorCode 一覧を参照してください。**

PrintTwoNormal メソッド

形式

void PrintTwoNormal (PrinterStation *Stations*, string *Data1*, string *Data2*);

パラメータ	説明
Station	使用される POS プリンターステーション。
Data1	第 1 のステーションに印刷される文字。
Data2	第 2 のステーションに印刷される文字。

説明

2 つの文字列を 2 つの POS プリンターに同時に印字する時にこのメソッドを呼び出します。

このメソッドはスリッププリンターを対象としているため、本クラスライブラリではサポートされません。

例外

このメソッドを呼び出したときに、次の PosControlException をスローすることがあります。

値	意味
Illegal(106)	指定した POS プリンター（レシート以外）が存在しません。
その他	**ErrorCode 一覧を参照してください。**

RotatePrint メソッド

形式

void RotatePrint (PrinterStation *Station*, PrintRotation *Rotation*);

パラメータ	説明
Station	Receipt(2)を指定
Rotation	回転方向。下記の値を参照。

Rotation の値は、以下の通りです。

値	意味
Right90(257)	右に 90 度回転（時計回りに）印刷の開始。
Left90(258)	左に 90 度回転（時計と反対回りに）印刷の開始。
Rotate180(259)	180 度回転印刷、つまり、倒立印字の開始。
Normal(1)	回転印刷の終了。

説明

このメソッドは、**AsyncMode** が **FALSE** ならば、同期的に実行され、**AsyncMode** が **TRUE** ならば、非同期的に実行されます。

もし、*Rotation* が Rotate180 ならば、倒立印刷モードが開始されます。**RotatePrint** が *Rotation* パラメータを Normal に設定して呼ばれるまで、**PrintNormal** と **PrintImmediate** の呼び出しによるデータは倒立して印刷されます。行は POS プリンターコントロールに送られた順に印刷され、各行の先頭はプリンターの右マージンに合わされます。**PrintNormal** と **PrintImmediate** の印刷メソッドが、倒立印刷モード中に使用されます

Rotation が Right90、Left90 ならば、横書きモードが開始されます。**RotatePrint** が *Rotation* パラメータを Normal に設定して呼ばれるまで、**PrintNormal** メソッドの呼び出しによるデータはバッファリングされます。（この場合、上記のメソッドのデータはバッファリングされるだけで、印刷は開始しません。また、**AsyncMode** プロパティの値は、そのオペレーションに影響しません。つまり、その要求に **OutputID** は割り当てられませんし、**OutputCompleteEvent** も通知しません。尚、この場合、POS プリンターのエラー状態に関わらず、各メソッドは成功します。例えば、POS プリンターの電源が切れていても、**RotatePrint** による印刷データのバッファリング中は、各メソッドの呼びだし時にエラーを返しません。）

横書きモード時は **PrintNormal** メソッドの呼び出しによるバッファリングされる文字データによって、モノクロは 0～1662dot 2 色時は 0～831dot の範囲で横幅を自動設定します。本クラスライブラリは、バッファリング中の文字データを解析し、すべての行における幅の最大値にあわせて、横幅が決定されます（下記の表を参考）。 ただし、文字データ内にエスケープシーケンスによるビットマップ印刷やバーコード印刷が指定された場合には横幅の算入が行われないため他の文字データで計算された横幅に収まらないビットマップやバーコードの印刷は正常に行われません。

文字データ総数の幅が 1662dot(2 色時は 831dot)を超えた場合は、横幅は 1662dot(2 色時は 831dot)となり、はみ出したデータについては、ページ内に折り返して印字されます。また、文字データの横幅がエスケープシーケンスによって 2 倍以上になる場合も、倍率をかけた値で計算されます。（例:フォント A　ANK 文字が 横 2 倍で指定された場合は、24 dot として計算します。）
尚、バッファリングされたデータが無い場合（**PrintNormal** メソッドが一度も実行されていない場合）は、何も印字を行いません。

文字 1 文字あたりの横幅(dot)

フォント(RecLineChars プロパティを参照してください。)	ANK	漢字
フォント A	12 dot	24 dot
フォント B	10 dot	20 dot
フォント C	8 dot	16 dot

倒立印刷モードで **PrintBitmap, PrintMemoryBitmap** メソッドを発行しても、ビットマップは倒立印刷されません。
倒立印刷モードで **SetBitmap** を行ったビットマップも、倒立しないで登録されます。

Rotation に BarCode、あるいは Bitmap が含まれている場合には、（**PrintBarCode** で印刷される）バーコード、あるいは（**PrintBitmap**、または'ESC|#B'エスケープシーケンスで印刷される）ビットマップはすべて、**RotatePrint** で回転印刷できます。回転方向は、それぞれ **RecBarCodeRotationList**、**RecBitmapRotationList** プロパティで制限されます。
Rotation に Barcode が含まれている場合には、**RotateSpecial** の内容は無視されます。

もし、*Rotation* が Normal ならば、回転印刷モードは終了します。もし、横向き回転印刷モードが有効な時に、あるデータが **PrintNormal** によってバッファリングされていたならば、そのバッファリングされたデータは印刷されます。全体の回転した行のブロックは、一つのメッセージとして扱われます。

ClearOutput メソッドを呼ぶことにより、回転印刷モードは中止されます。いかなるバッファリングされた横向き回転印刷行も削除されます。

尚、エスケープシーケンスで縦 n 倍を指定し、回転印刷を行った場合、印字が重なったり、あるいははみ出る場合があります。これらの場合は縦 n 倍のデータの前に改行コードを入れてください。（例：縦 3 倍角のデータを出力するには、2 つの改行コードを入れておくことで正常に印字されます。基本的に縦 n 倍の場合,n-1 個の改行コードを挿入すれば問題ありません。）

例外

このメソッドを呼び出したときに、次の PosControlException をスローすることがあります。

値	**意味**
例外なし	メソッドは正常終了しました。
Illegal(106)	指定した POS プリンター（レシート以外）が存在しません。 または、*Station* が指定した回転をサポートしていません。 異なる回転モード中です。一旦、Normal(1)を指定するか、**ClearOutput** にて回転印字をクリアした後、再施行してください。 ページモード開始中です。ページモードを終了後、再施行してください。
NoHardware(107)	POS プリンターはオフ、またはオフラインです。
Failure(111)	本クラスライブラリはエラー状態です。エラー状態を解除して実行してください。
Busy(113)	出力中なので実行できません。（**AsyncMode** が **FALSE** の時のみ返されます。）
Extended(114)	**ErrorCodeExtended** = CoverOpen(201): POS プリンターカバーが開いています。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Recempty(203):レシート用紙がありません。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Blackmark(10001):ブラックマークエラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Fatal (10003):フェイタルエラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Overheat (10006):ヘッド昇温エラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。）
その他	**ErrorCode 一覧**の項目を参照してください

SetBitmap メソッド

形式

void SetBitmap (int *BitmapNumber*, PrinterStation *Station*, string *FileName*, int *Width*, int *Alignment*);

パラメータ	説明
BitmapNumber	このビットマップに割り当てられる番号。 有効な値は、1～20 までの値です。
Station	Receipt(2)を指定
FileName	Windows ビットマップファイル名。ファイルは圧縮されたフォーマットではいけません。 （フルパスまたは相対パス指定） 空文字が設定された場合、指定された *BitmapNumber* のビットマップを POS プリンターから消去します。
Width	ビットマップの印刷幅。値は **PrintBitmap** 参照。
Alignment	ビットマップの印刷位置。値は **PrintBitmap** 参照。

説明

まもなく印刷するビットマップに関する情報を保存する時に呼び出します。

ビットマップは、印刷データ内にビットマップ印刷のエスケープシーケンスをもつ **PrintNormal** あるいは **PrintImmediate** を呼ぶことによって印刷されます。

２色印刷設定の場合、黒を第一色、その他を第二色で保存します。

登録可能なビットマップは、横サイズ(*Width*)が **RecLineWidth** dot 以下（*Alignment* が絶対位置指定の場合、*Width*+ *Alignment*<= **RecLineWidth**）、且つ縦サイズが 2304dot 以下となり、且つディザリング後（POS プリンターに対して解釈可能なビットマップデータに変換した後）のデータサイズが、384KByte 以下のデータである必要があります。この条件に適合しない場合、 Toobig(206)が通知されます。また、不揮発メモリ上に空き容量が存在しない場合についても、Toobig(206)が通知されますので、この場合、*FileName* パラメータに空白を指定して、POS プリンターからビットマップデータを削除して空き容量を確保した上で、再度実行してください。

２色印字設定の場合は、24 ビットカラーのビットマップデータを登録することは出来ません。

※本クラスライブラリでは、**SetBitmap** を実行すると、ビットマップを POS プリンターの不揮発メモリ上に書き込む為、いったん本クラスライブラリの **Release** を行った後でも、セットしたビットマップ自体は有効となります。つまり、1 度設定を行えば、エスケープシーケンスによるビットマップの印字が有効となります。

※指定されたビットマップデータがモノクロの場合、モノクロビットマップがプリンターに設定されます。モノクロ以外のデータの場合、**CapRec2Color** プロパティが TRUE の場合は 2 色ビットマップ印字が実行され、FALSE の場合はモノクロデータとして印字されます。

※メソッドのパラメータに Left(-1), Center(-2), Right(-3)で登録したビットマップを **RotatePrint** メソッドによる左右 90 度回転印刷時すると、標準(縦)方向の印字幅基準での左寄せ、中央寄せ、右寄せになります。

登録可能な画像形式は PrintBitmap プロパティを参照して下さい。

例外

このメソッドを呼び出したときに、次の PosControlException をスローすることがあります。

値	意味
例外なし	メソッドは正常終了しました。
Illegal(106)	指定した POS プリンター（レシート以外）が存在しません。 ・*BitmapNumber* が不正な値です。 ・POS プリンター（レシート以外）が存在しません。 ・*Station* がビットマップ印刷をサポートしていません。 ・*Width* が大きすぎます。 ・*Alignment* が、不正な値であるか、大きすぎます。
Noexist(109)	*FileName* で指定されたビットマップファイルが見つかりませんでした。
Failure(111)	POS プリンターにビットマップデータを送信できませんでした。カバーが開いているか、レシート用紙がないか、POS プリンターの電源が切れてる可能性があります。
Busy(113)	デバイスが出力中のため、実行できません。
Extended(114)	**ErrorCodeExtended** =Toobig(206): ビットマップが、変換無しで印刷するにはその幅が広すぎるか、あるいは変換するには大きすぎます。 **ErrorCodeExtended** = Badformat(207): 指定されたファイルが、ビットマップファイルでないか、あるいはサポートされていないフォーマットです。
その他	**ErrorCode 一覧を参照してください。**

SetLogo メソッド

形式

void SetLogo (PrinterLogoLocation *Location*, string *Data*);

パラメータ	説明
Location	設定するロゴ。Top(1)あるいは Bottom(2)です。
Data	ロゴを作る文字。プリント可能な文字とエスケープシーケンス、キャリッジリターン(13 decimal)、ニューライン/ラインフィード(10 decimal)から成ります。

説明

データ文字列を上端あるいは下端のロゴとして保存する時に呼び出します。

プリントデータ内に、上端ロゴあるいは下端ロゴのエスケープシーケンスを含んだ **PrintNormal,**あるいは **PrintImmediate** メソッドを呼ぶことによって、ロゴは印刷されます。

例外

このメソッドを呼び出したときに、次の PosControlException をスローすることがあります。

値	意味
例外なし	メソッドは正常終了しました。
Busy(113)	出力中なので実行できません。
Illegal(106)	不正な *Location* が指定されました。
その他	**ErrorCode 一覧を参照してください。**

TransactionPrint メソッド

形式

void TransactionPrint(PrinterStation *Station*, PrinterTransactionControl *Control*);

パラメータ	説明
Station	Receipt(2)を指定
Control	一括処理。下記の値を参照。

Control の値は、以下の通りです。

値	意味
Transaction(11)	一括処理の開始。
Normal(12)	バッファデータを印字して一括処理を終了する。

説明

一括処理モードに入る時または抜ける時にこのメソッドを呼び出します。

Control が Transaction(11)ならば、一括処理モードに入ります。その後の **PrintNormal**、**CutPaper**、**RotatePrint**、**PrintBarCode**、**PrintBitmap** メソッドへの呼び出しは、*Control* パラメータに Normal(12) をセットして **TransactionPrint** が呼び出されるまで、印刷データをサービスオブジェクトでバッファリングします。（この場合、前記のメソッドの印刷データはバッファリングされるだけで、印刷は開始しません。また、**AsyncMode** プロパティの値は、そのオペレーションに影響しません。つまり、その要求に **OutputID** は割り当てられませんし、**OutputCompleteEvent** も通知しません。尚、この場合、POS プリンターのエラー状態に関わらず、各メソッドは成功します。例えば、POS プリンターの電源が切れていても、**TransactionPrint** による印刷データのバッファリング中は、各メソッドの呼びだし時にエラーを返しません。）

Control が Normal(12)ならば、一括処理モードを抜けます。**PrintNormal**、**CutPaper**、**RotatePrint**、**PrintBarCode**、**PrintBitmap** メソッドによりデータがバッファされていたならば、そのデータは印刷されます。一括処理全体は一つのメッセージとして処理されます。このメソッドは **AsyncMode** が **FALSE** ならば、同期的に実行され **AsyncMode** が **TRUE** ならば、非同期的に実行されます。

ClearOutput メソッドを呼び出すことによって、一括処理モードは取り消されます。バッファされた印刷行も削除されます。

注意点として、**RotatePrint** メソッドを実行した場合があります。**TransactionPrint** メソッドを実行し、一括処理モードを抜けるまで、**RotatePrint** メソッドを Right90(257)で呼び出し、**PrintNormal** メソッド呼び出し、**RotatePrint** メソッドを Normal (1)で呼び出しによる印字はされません。また、**RotatePrint** メソッドを Right90(257) で呼び出し、**TransactionPrint** メソッド Transaction(11) で呼び出した場合、**TransactionPrint** メソッドによるバッファリングの方が優先度が高い為、この間にバッファリングされたデータは、回転印字せず、正しく印字することができません。よって、**RotatePrint** メソッドを実行する場合は、**TransactionPrint** メソッドの後に実行してください。

一括処理モード中、ページモード機能を利用することはできません。

一括処理モード開始中に再度一括処理モードを開始した場合、それまでバッファリングした印刷データは破棄されます。

例外

このメソッドを呼び出したときに、次の PosControlException をスローすることがあります。

値	**意味**
例外なし	メソッドは正常終了しました。
Illegal(106)	指定した POS プリンター（レシート以外）が存在しません。 ページモード機能開始中です。
NoHardware(107)	POS プリンターはオフ、またはオフラインです。
Failure(111)	本クラスライブラリはエラー状態です。エラー状態を解除して実行してください。
Busy(113)	出力中なので実行できません。（**AsyncMode** が **FALSE** で Control パラメータが　Normal(12)の時のみ返されます。）
Extended(114)	**ErrorCodeExtended** = CoverOpen(201): POS プリンターカバーが開いています。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Recempty(203):レシート用紙がありません。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Blackmark(10001):ブラックマークエラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Fatal (10003):フェイタルエラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。） **ErrorCodeExtended** = Overheat (10006):ヘッド昇温エラーが発生しました。（**AsyncMode** が **FALSE** の時のみ返されます。）
その他	**ErrorCode 一覧を参照してください。**

ValidateData メソッド

形式

void ValidateData(PrinterStation *Station*, string *Data*);

パラメータ	説明
Station	Receipt(2)を指定
Data	判定されるデータ。印刷可能データとエスケープシーケンスを含みます。 詳細は **BinaryConversion** プロパティを参照してください。

説明

PrintNormal,**PrintImmediate** メソッドを呼び出す前に、指定された POS プリンターにとって、一つ以上のエスケープシーケンスを含むデータシーケンスが、有効か否かを判定する時に呼び出します。

このメソッドは、全く印刷しませんが、その POS プリンターの能力を判定するために使用されます。

Illegal(106)が返されるケース 又は Failure(111)が返されるケースに記載の無いエスケープシーケンスは、判定せずに全て例外なしを返します。

例外

このメソッドを呼び出したときに、次の PosControlException をスローすることがあります。

値	意味
例外なし	データは有効です。
Illegal(106)	一つ以上のエスケープシーケンスが範囲外です。しかし、コントロールは有効な選択肢を選ぶことができます。あるいは、対象となるステーションが未サポートです。
Failure(111)	一つ以上のエスケープシーケンスがサポートされていません。選ぶことのできる選択肢はありません。
その他	**ErrorCode 一覧を参照してください。**

Illegal(106)が返されるケースは、以下の場合です。

エスケープシーケンス	状況
用紙カット/フィードと用紙カット	パーセンテージ'#'はサポートされていません。（1～100 のみ有効）
アンダーライン	太さ'#'はサポートされていません。(1～2 のみ有効)
縦倍率	倍率'#'はサポートされていません。（1～8 のみ有効）
横倍率	倍率'#'はサポートされていません。（1～8 のみ有効）

Failure(111)が返されるケースは、以下の場合です。

エスケープシーケンス	状況
フィードとカットとスタンプ印刷	サポートされていません。
スタンプ印刷	サポートされていません。
ビットマップ印刷	ビットマップ番号'#'が範囲外です。(1～20 のみ有効)
逆フィード	サポートされていません。
カスタムカラー	サポートされていません。
赤色	**CapRec2Color** が FALSE の時、未サポート。
網掛け文字	サポートされていません。
サブスクリプト/スーパースクリプト	サポートされていません。

DrawRuledLine メソッド

形式

void DrawRuledLine (PrinterStation *Station*, **string** *PositionList*, **LineDirection** *Direction*, **int** *LineWidth*, LineStyle *Style*, **int** *LineColor***);**

例外

このメソッドを呼び出したときに、次の PosControlException をスローすることがあります。

値	意味
Illegal(106)	このメソッドはサポートされません。

4．7．イベント

DirectIOEvent イベント

形式

DirectIOEventHandler DirectIOEvent;

パラメータ	説明
int EventNumber	イベント番号。サービスオブジェクトが割り当てた特定値です。
int Data	数値データ。イベント番号とサービスオブジェクトで変化する特定値です。
object Object	バイト列のオブジェクト。データは変換されずに Bstring 一文字は 1 バイトで格納されます。BinaryConversion=NIBBLE として動作します。

説明

アプリケーションと直接通信するために、サービスオブジェクトが通知します。

プリンターからの通常ステータス通知以外のデータを受信した場合に、以下の *EventNumber* で 1 バイトずつ通知します。

EventNumber には、 NotAsb(=101)が設定され、*Data* にバイトデータを 10 進数変換した値(0 ～ 255) が設定されます。

ErrorEvent イベント

形式

DeviceErrorEventHandler ErrorEvent;

パラメータ	説明
ErrorCode ErrorCode	エラーイベントを起こした要因のコード。 値は ErrorCode の項目を参照。
int ErrorCodeExtended	エラーイベントを起こした要因の拡張コード。下記の値を参照。
ErrorLocus ErrorLocus	Output(1)に設定します。非同期出力中にエラーが起こりました。
ErrorResponse ErrorResponse	エラーイベント応答です。下記の値を参照。

もし, ErrorCode が Extended(114)であれば、ErrorCodeExtended は次に示す値のいずれかに設定されます。

値	意味
Coveropen(201)	POS プリンターカバーが開いています。
RecEmpty(203)	レシート用紙がありません。
Fatal (10003)	POS プリンターでフェイタルエラーが発生しました。
Overheat (10006)	プリンターでヘッド昇温が発生しました。
Cutterjam (10008)	カッタージャムエラーが発生しました。

ErrorResponse によって指定される位置の内容は、Retry(11)のデフォルト値にプリセットされます。
アプリケーションは、次に示す値の内の一つを設定します。

値	意味
Retry(11)	その非同期処理を再度試みます。エラー状態は抜けます。
Clear(12)	その非同期出力を含むすべてのバッファリングされたデータを削除します（**ClearOutput** メソッドと同様の効果があります）。エラー状態は抜けます。

説明

非同期で実行可能なメソッド実行時に、本クラスライブラリの状態がエラー状態に移った時に通知します。

OutputCompleteEvent イベント

形式

OutputCompleteEventHandler OutputCompleteEvent;

OutputID パラメータは、完了した非同期出力リクエストの ID 番号を示します。

説明

前に開始された非同期出力リクエストが正常に終了したときに通知します。

StatusUpdateEvent イベント

形式

StatusUpdateEventHandler StatusUpdateEvent;

説明

Status は、次に示す値のいずれかに設定されます。

値	意味
CoverOpen(11)	POS プリンターカバーが開いています。
CoverOK(12)	POS プリンターカバーが閉じています。
RecEmpty(24)	レシート用紙がありません。
RecNearempty(25)	レシート用紙がニアエンドです。
RecPaperok(26)	レシート用紙が用意できました。
RecCoverOpen(62)	カッタージャムエラーが発生しました。
RecCoverOK(63)	カッタージャムエラーが復旧しました。
Idle(1001)	全ての非同期出力が、成功するか、または削除されることによって終了しました。POS プリンターの **State** は現在_dle(2)です。**FlagWhenIdle** プロパティは、このイベントが通知するために **TRUE** でなければなりません。そして、POS プリンターコントロールは、イベントが通知する前に、そのプロパティを **FALSE** へ自動的にリセットします。
PowerOnline(2001)	デバイスは電源オンでかつレディ状態です。（**PowerNotify** = Enabled(1)の場合に通知）
PowerOffOffline(2004)	デバイスは電源オフ、またはオフライン状態です。（**PowerNotify** = Enabled(1)の場合に通知）
Progress(2100) + 1 to 100 (1 ～ 100 は進捗率)	ファームウェア進捗率を設定します。
Complete(2200)	ファームウェア更新に成功しました。
FailedDevOK(2201)	ファームウェア更新処理が失敗しましたが、デバイスはまだ操作可能です。

説明

プリンターデバイス側に重要な状態の変化が起こった時に通知されます。

デバイスがイネーブルとなったとき、アプリケーションにデバイスの状態を知らせるために、コントロールは最初の **StatusUpdateEvent** を発行します。

参照

CapPowerReporting プロパティ、**PowerNotify** プロパティ

5. OPOS インターフェース仕様（ドロワー）

5．1．一覧

プロパティ

共通	型	アクセス	初期化	初期値･書込み条件
CapCompareFirmwareVersion	bool	R	Open	FALSE
CapPowerReporting	PowerReporting	R	Open	None(0)
CapStatisticsReporting	bool	R	Open	FALSE
CapUpdateStatistics	bool	R	Open	FALSE
CheckHealthText	string	R	Open	“”
Claimed	bool	R	Open	FALSE
DeviceEnabled	bool	R/W	Open	FALSE Open 後、書込み可能
FreezeEvents	bool	R/W	Open	FALSE Open 後、書込み可能
OpenResult	int	R	無し	0
PowerNotify	PowerNotification	R/W	Open	Disabled(0) 書込み不可能
PowerState	PowerState	R	Open	Unknown(2000)
State	ControlState	R	--	1
ControlObjectDescription	string	R	--	“FP CashDrawer Control Object (C) Fujitsu Isotec”
ControlObjectVersion	Version	R	--	1013XXX
ServiceObjectDescription	string	R	Open	“Fujitsu Isotec FP CashDrawer Service Object”
ServiceObjectVersion	int	R	Open	1013XXX
DeviceDescription	string	R	Open	“FP CashDrawer (C)20xx Fujitsu Isotec”
DeviceName	string	R	Open	Open 時にパラメータに設定する名前

専用	型	アクセス	初期化	初期化
CapStatus	bool	R	Open	初期値は PosExplorer 外部設定値の"CapStatus"の値
CapStatusMultiDrawerDetect	bool	R	Open	FALSE
DrawerOpened	bool	R	Open & Enable	FALSE

※アクセスの R は読込み専用、R/W は読込み、書き込み可能であることを示します。また、初期化は、初期化に必要なメソッド・プロパティを示し、Open は Open メソッド、Claim は Claim メソッド、Enable は DeviceEnabled プロパティを TRUE に設定することを意味します。初期化に必要な手続きを実行しない場合、ErrorCode プロパティにエラーが設定される場合があります。また、初期化条件が Open&Claim および Open,Claim&Enable 状態になっているプロパティについては、Open メソッド実行後に取得可能となりますが、Open,Claim &Enable 後にはじめて値が初期化される場合があるため、プロパティ取得時には条件を満たした上でアクセスを行なってください。

メソッド

共通	必要条件
Open	--
Close	Open
Claim	Open
Release	Open & Claim
CheckHealth	Open & Enable
CompareFirmwareVersion	Open & Enable
DirectIO	Open
ResetStatistics	Open & Enable
RetrieveStatistics	Open & Enable
UpdateFirmware	Open & Enable
UpdateStatistics	Open & Enable

専用	必要条件
OpenDrawer	Open & Enable
WaitForDrawerClose	Open & Enable

イベント

名称	必要条件
DirectIOEvent	Open & Enable
StatusUpdateEvent	Open & Enable

5．2．共通プロパティ

ドロワーに対して共通に提供されるプロパティを説明します。
プロパティは読込み専用のものと、読込み、書き込み可能なものの 2 種類が存在します。書き込み可能なプロパティには、R/W をプロパティ名の横に記述します。
また、戻り値は特別な意味合いがあるものについてのみ明記します。初期化条件を満たさずにアクセスした場合の例外については、ErrorCode プロパティを参照してください。

CapCompareFirmwareVersion プロパティ

形式

bool CapCompareFirmwareVersion;

説明

TRUE ならばサービス/デバイスは、ファームウェアファイルのバージョンと物理デバイス内のファームウェアバージョンを比較する機能を有します。
このプロパティは **Open** メソッドにより **FALSE** に初期化されます。

CapPowerReporting プロパティ

形式

PowerReporting CapPowerReporting;

説明

デバイスの電源通知能力を識別します。電源通知能力を示す値は下記の通りです。

値	意味
None(0)	SO はデバイスの電源状態の判断ができません。よって、電源通知機能は機能しません。
Standard(1)	SO は、OffOffLine（電源オフまたはオフライン）と Online の 2 種類の電源状態を判断し通知が可能です。
Advanced(2)	SO は、Off、OffLine、OnLine の 3 種類すべての電源状態を判断し通知が可能です。

このプロパティは **Open** メソッドにより初期化されます。

CapStatisticsReporting プロパティ

形式

bool CapStatisticsReporting;

説明

このプロパティは **Open** メソッドにより **FALSE** に初期化されます。スタティスティクスの通知はサポートしません。

CapUpdateStatistics プロパティ

形式

bool CapUpdateStatistic;

説明

このプロパティは **Open** メソッドにより **FALSE** に初期化されます。スタティスティクスの通知はサポートしません。

CheckHealthText プロパティ

形式

string CheckHealthText;

説明

直前に呼び出した **CheckHealth** メソッドの結果を保持します。以下に診断結果を示します。

"Internal Hcheck: Successful"

内部チェックに成功しました。

"External HCheck : Successful"

外部チェックに成功しました。

"External HCheck : Failure"

外部チェックに失敗しました。

"Interactive HCheck : Not Supported"

対話チェックはサポートしません。

最初の **CheckHealth** メソッド呼び出し以前にはこの値は初期化されません。（空文字）

Claimed プロパティ

形式

bool Claimed;

説明

TRUE : デバイスの排他アクセス権が獲得されています。

FALSE : デバイスはほかのアプリケーションと共有できるよう解放されています。

多くの場合、デバイスの排他アクセス権を獲得した後で、メソッドやプロパティへのアクセス、イベントの発生が可能となります。

Claimed プロパティの値は **Open** メソッドによって **FALSE** に初期化されます。

DeviceDescription プロパティ

形式

string DeviceDescription;

説明

"FP CashDrawer (C)20xx- Fujitsu Isotec" が設定されています。

このプロパティはデバイスを識別する文字列で、デバイスとそれに関連する情報を示します。

このプロパティは **Open** メソッドによって初期化されます。

DeviceEnabled プロパティ R/W

形式

bool DeviceEnabled;

説明

TRUE :

デバイスはイネーブルにされています（動作状態です）。**TRUE** に変更するとイネーブルにされます。

FALSE :

デバイスはディセーブルにされています。FALSE に変更すると、可能ならば物理的にディセーブルにされます。後続の入力は廃棄され、出力動作を行うことはできません。デバイスを使用する前に、アプリケーションはこのプロパティを **TRUE** に設定しなければなりません。

このプロパティは **Open** メソッドにより **FALSE** に初期化されます。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
例外なし	プロパティの設定に成功しました。
Failure(111)	接続ポートのオープンに失敗しました。他のプログラムから接続ポートが使用されていないか、接続ポートが存在しているかを確認した後、もう一度実行してください。
NoHardware(107)	POS プリンターがオフ／オフライン、またはケーブルが接続されていません。これらを解消した後、もう一度実行してください。 ※USB インターフェースの場合、POS プリンターを接続していても、POS プリンターに設定されているシリアル番号とレジストリに設定されたシリアル番号が異なる場合は、この例外が発生します。 ※LAN インターフェースの場合、POS プリンターが接続されていなくても、例外なしが返ります。

DeviceName プロパティ

形式

string DeviceName;

説明

"FP2x00SERDR1"," FP2x00SERDR2", "FP2x00USBDR1", "FP2x00USBDR2", "FP2x00LANDR1_xxx.xxx.xxx.xxx", "FP2x00LANDR2_xxx.xxx.xxx.xxx" (xxx.xxx.xxx.xxx: IP アドレス)

のいずれかが設定されます。設定される値は、**Open** したデバイスによって異なります。このプロパティは、デバイスとそれに関連する情報を示します。**DeviceDescription** プロパティの短縮バージョンで、30 文字までに制限されています。

このプロパティは **Open** メソッドにより初期化されます。

FreezeEvents プロパティ R/W

形式

bool FreezeEvents;

説明

TRUE の場合、コントロールからイベントが通知されません。

凍結が解除されるまで、イベントはコントロールが保持しています。

FALSE の場合、コントロールからイベントが通知されます。凍結されていた間に保持されたイベントがあれば、**FreezeEvents** を **FALSE** に変更すると、そのイベントが通知されます。

イベントによる割り込みが望ましくない状況で、アプリケーションはイベントの凍結を選択できます。

このプロパティは **Open** メソッドにより **FALSE** に初期化されます。

PowerNotify プロパティ R/W

形式

PowerNotification PowerNotify;

説明

アプリケーションから設定された電源通知機能のタイプです。

電源通知機能を示す値は下記のとおりです。

値	意味
Disabled(0)	コントロールはアプリケーションに対していかなる電源通知をも提供しません。電源通知に関する **StatusUpdateEvent** は通知されず、また **PowerState** プロパティにも何も設定されません。
Enabled(1)	

このプロパティは **Open** メソッドにより Disabled(0)に初期化されます。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
Illegal(106)	本プロパティを設定することはできません。

PowerState プロパティ

形式

PowerState PowerState;

説明

判断が可能であれば現在のデバイスの電源状態が設定されます。

電源状態を示す値は下記のとおりです。

値	意味
unknown(2000)	デバイスの電源状態の判断ができません。
Online(2001)	
Off(2002)	
Offline(2003)	
OffOffline(2004)	

ServiceObjectDescription プロパティ

形式

string ServiceObjectDescription;

説明

デバイスをサポートしているサービスオブジェクトと、そのデバイスを製造している会社を示す文字列です。
"Fujitsu Isotec FP CashDrawer Service Object"と設定されています。
このプロパティは **Open** メソッドによって初期化されます。

ServiceObjectVersion プロパティ

形式

System.Version ServiceObjectVersion;

説明

"1013XXX" が設定されています。サービスオブジェクトバージョン番号を示します。（XXX は、サービスオブジェクトの提供時期によって異なります。）このプロパティは **Open** メソッドによって初期化されます。

State プロパティ

形式

ControlState State;

説明

コントロールの現在の状態を示します。

値	意味
Closed(1)	コントロールはクローズしています。（デフォルト）
Idle(2)	コントロールは正常な状態にあり、ビジーではありません。
Busy(3)	コントロールは正常な状態にあり、出力を実行しているためビジーです。
Error(4)	エラーが報告され、通常の I/O を再開するには、その前にアプリケーションがコントロールを正常な状態に戻さなければなりません。

このプロパティはいつでも読取り可能です。

5．3．共通メソッド

CheckHealth メソッド

形式

string CheckHealth (HealthCheckLevel *Level*);

Level パラメータは、デバイスで実行するヘルスチェックのタイプを示します。以下の値を指定できます。

値	意味
Internal(1)	デバイスを物理的に使用しないヘルスチェックを実行します。常に Success を返します。
External(2)	デバイスを使用して完全なテストを実行します。ドロワーを可能ならばオープンします。オープンに成功した場合は、Success が返ります。他のアプリケーションから排他アクセスがされている場合、本メソッドは失敗します。
Interactive(3)	デバイスとの対話テストを実行します。サポートされません。

説明

デバイスの状態をテストするときに呼び出します。このメソッドの結果は **CheckHealthText** プロパティに格納されます。**CheckHealth** メソッドは常に同期です。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
例外なし	ヘルスチェックプロシージャが適切に開始されたことを示し、確認できた場合、デバイスが正常であることを示します。しかし、正常かどうかはテスト結果を見ないと決定できません。
Claimed(102)	他のデバイスが排他アクセスを行なっています。
Illegal(106)	サポートされていない *Level* パラメータが指定されました。
NoHardware(107)	ヘルスチェックプロシージャに失敗しました。ドロワーが接続されている POS プリンターがオフになっているかまたは、オフラインになっています。External(2)が設定された場合のみ、格納されます。
その他	**ErrorCode 一覧を参照してください。**

Claim メソッド

形式

int Claim (int *Timeout*);

Timeout パラメータは、排他アクセス権を獲得するまでの最大待ち時間（ミリ秒単位）を示します。

ゼロの場合、メソッドはデバイスの排他アクセスが獲得できなかった場合でも、その結果を直ちに戻します。

Forever(-1)が設定されている場合は、メソッドは排他アクセス権が獲得できるまで必要なだけ待ちます。

説明

デバイスに対して排他アクセスを要求するときに、このメソッドを呼び出します。

ドロワーデバイスは共有可能デバイスなので、排他アクセス権の取得は必須ではありません。

成功すると、**Claimed** プロパティは **TRUE** に設定されます。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
例外なし	排他アクセス権が認められ、処理可能なデバイス接続が確立されました。**Claimed** プロパティは **TRUE** になっています。このアプリケーションがすでにデバイスを排他アクセスしていた場合も戻されます。
Illegal(106)	無効な ***Timeout*** パラメータが指定されています。
Timeout(112)	別のアプリケーションがデバイスに排他アクセスしていて、解放されるのを待っていましたが ***Timeout*** 時間（ミリ秒単位）が過ぎました。もしくは、***Timeout*** 時間（ミリ秒）経過しても、デバイスが処理可能な状態になりませんでした。
その他	**ErrorCode 一覧を参照してください。**

Close メソッド

形式

void Close ();

説明

デバイスとそのリソースを解放するときに呼び出します。

DeviceEnabled プロパティが **TRUE** の場合、まずデバイスがディセーブルにされます。

Claimed プロパティが **TRUE** の場合、まずデバイスの排他アクセスが解除されます。

イベント処理中（イベントハンドラ内）では実行しないでください。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
例外なし	デバイスがディセーブルにされ、クローズしました。
その他	**ErrorCode 一覧を参照してください。**

DirectIO メソッド

形式

DirectIOData DirectIO (int *command*, int *data*, object *obj*);

説明

サービスオブジェクトと直接通信する時に呼び出します。

本メソッドはサポートされません。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
Illegal(106)	このメソッドは有効ではありません
その他	**ErrorCode 一覧を参照してください。**

Open メソッド

形式

void Open ();

説明

デバイスをオープンするときに呼び出します。

Open メソッドが成功すると、共通プロパティ、その他のクラス固有プロパティが初期化されます。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
Illegal(106)	該当のコントロールはすでにオープンしています。

Release メソッド

形式

void Release ();

説明

デバイスの排他アクセスを解除するときにこのメソッドを呼び出します。

DeviceEnabled プロパティが **TRUE** で排他デバイスならば、デバイスはディセーブルにされます。

イベント処理中（イベントハンドラ内）では実行しないでください。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
例外なし	排他アクセスが解除されました。**Claimed** プロパティは **FALSE** になります。
Illegal(106)	アプリケーションは該当のデバイスへの排他アクセス権を持っていません。
その他	**ErrorCode 一覧を参照してください。**

ResetStatistics メソッド

形式

void ResetStatistics ();

説明

このメソッドはサポートされていません。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。s

値	意味
Illegal(106)	このメソッドはサポートされません。

RetrieveStatistics メソッド

形式

sting RetrieveStatistics ();

説明

このメソッドはサポートされていません。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
Illegal(106)	このメソッドはサポートされません。

形式

string RetrieveStatistics (StatisticCategories *statistics*);

説明

このメソッドはサポートされていません。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
Illegal(106)	このメソッドはサポートされません。

形式

string RetrieveStatistics (string [] *statistics*);

説明

このメソッドはサポートされていません。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
Illegal(106)	このメソッドはサポートされません。

UpdateFirmware メソッド

形式

void UpdateFirmware (String *FirmWareFileName*);

説明

本メソッドはサポートしません。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
Illegal(106)	このメソッドはサポートされません。

UpdateStatistics メソッド

形式

void UpdateStatistics (Statistic [] *statistics*);

説明

本メソッドはサポートしません。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
Illegal(106)	このメソッドはサポートされません。

形式

void UpdateStatistics (StatisticCategories *statistics*, object *value*);

説明

本メソッドはサポートしません。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
Illegal(106)	このメソッドはサポートされません。

5．4．専用プロパティ

CapStatus プロパティ

形式

bool CapStatus;

説明

TRUE：ドロワーの開閉状態を通知することができます。

FALSE：ドロワーの開閉状態を通知することができません。

このプロパティは **Open** メソッドにより初期化されます。

“FPxxx[デバイス名]DR1”で定義される、１つめのドロワーのプロパティは、**TRUE** に、“FPxxx[デバイス名]DR2”で定義される２つめのドロワーのプロパティは、**FALSE** に設定されています。デバイス名には、”SER”(シリアルインターフェース), "USB"（USB インターフェース）, "LAN"（LAN インターフェース、デバイス名の後ろに IP アドレス xxx.xxx.xxx.xxx が付加されます）のいずれかが入ります

ただし、プロパティが **TRUE** であっても、ドロワーが接続されているプリンターに対して、プリンタークラスライブラリがイネーブル状態（**DeviceEnabeld**=**TRUE**）以外は、ドロワー開閉状態の通知はサポートされません。

CapStatusMultiDrawerDetect プロパティ

形式

bool CapStatusMultiDrawerDetect;

説明

FALSE : すべてのドロワーが閉められているか、１つ以上のドロワーが開けられています。複数ドロワー構成のどのドロワーに対しても、開閉状態を通知できるわけではありません。

このプロパティは、**Open** メソッドにより初期化されます。

DrawerOpened プロパティ

形式

bool DrawerOpened;

説明

TRUE：ドロワーは開けられています

FALSE：ドロワーは閉められています。

CapStatus プロパティが FALSE の場合、デバイスは状態変化の通知を行うことができず、この **DrawerOpened** プロパティは常に FALSE となっています。

このプロパティは、デバイスをイネーブルにしたときに適切な値に初期化されます。

※本プロパティは、**CapStatus** プロパティが **TRUE** であり、ドロワーが接続されているプリンターに対して、プリンタークラスライブラリがイネーブル状態（**DeviceEnabeld**=**TRUE**）以外は、ドロワー開閉状態の通知はサポートされません。

※“FPxxx[デバイス名]DR2”で定義される２つめのドロワーは、開閉状態を知ることができません。常に **FALSE** となります。デバイス名には、”SER”(シリアルインターフェース)、"USB"（USB インターフェース）のいずれかが入ります

5．5．専用メソッド

OpenDrawer メソッド

形式

void OpenDrawer ()

説明

ドロワーを開けます。他のアプリケーションから排他アクセスがされている場合、本メソッドは失敗します。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
例外なし	正常終了。
Claimed(102)	他のデバイスが排他アクセスを行なっています。
Failure(111)	デバイスと通信できませんでした。
Timeout(112)	既定時間内にドロワーが開いたことを検知できませんでした。ドロワーオープン・クローズを逆転して通知するドロワーの場合や、プリンターSO がデバイスオープンに失敗している等の原因で、ドロワーキックコマンド送信が完了しなかった場合に発生します。
その他	**ErrorCode 一覧を参照してください。**

WaitForDrawerClose メソッド

形式

void WaitForDrawerClose (int *BeepTimeout*, int *BeepFrequency*, int *BeepDuration*, int *BeepDelay*);

説明

本メソッドはサポートしません。

例外

このメソッドを呼び出したときに、次の **PosControlException** をスローすることがあります。

値	意味
Illegal(106)	このメソッドはサポートされません。

5．6．イベント

DirectIOEvent イベント

形式

DirectIOEventHandler DirectIOEvent;

説明

本メソッドは通知されません。

StatusUpdateEvent イベント

形式

StatusUpdateEventHandler StatusUpdateEvent;

説明

Status パラメータには、最新のドロワー状態が設定されています。

値	意味
Drawerclosed(0)	ドロワーは閉じています。
Draweropen(1)	ドロワーは開いています。

説明

ドロワーの開閉状態が変更されたときに通知されます。

6. PosExplorer の外部設定値について

以下に本クラスライブラリで使用する、外部設定値について下記に示します。

PosExplorer の設定値は手動で設定することも可能ですが、設定プログラムにて設定するようにしてください。また設定値は、本クラスライブラリが **Open** メソッドを実行したタイミングで読み込まれますので、本クラスライブラリ動作中に値を変更しても動作には反映されません。新しい設定値を反映させるには、一旦本クラスライブラリに **Close** メソッドを発行後、**Open** メソッドを発行してください。

7. ログファイルについて

FP POSPrinter クラスライブラリ、FP CashDrawer クラスライブラリは、PosExplorer 設定値の **LogFolder, LogFileName, LogLevel** を設定することで、ログファイルを出力します。ログファイル出力時の動作は下記の通りです。

1. 以下の命名規約でログファイルを作成します。

   LogFolder ="C:¥OPOS¥FIT¥FP_NET¥Log¥"
   LogFileName ="POSPrinter" と設定してある場合、
   ログファイルは、 **"C:¥OPOS¥FIT¥FP_NET¥Log¥POSPrinter_[HardwarePath]_yyyymmdd.log"** と設定され、ログファイルは実行される POS の日付が 2021 年 1 月 30 日だったならば、**"C:¥OPOS¥FIT¥FP_NET¥Log¥POSPrinter_[HardwarePath]_20210130.log"**というファイル名のログファイルが作成されます。
   ※ CashDrawer も同様です。

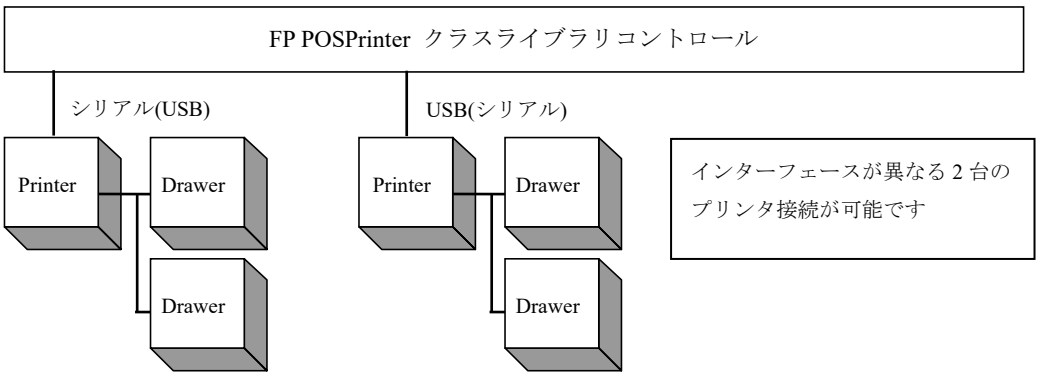
2. もし、1．の命名規約で作成されたファイルがすでに存在し、且つ現在の月とそのファイルの最新更新月が異なっている場合、そのログファイルを削除します。そうでない場合は、新規作成・または追加書込みとなります。

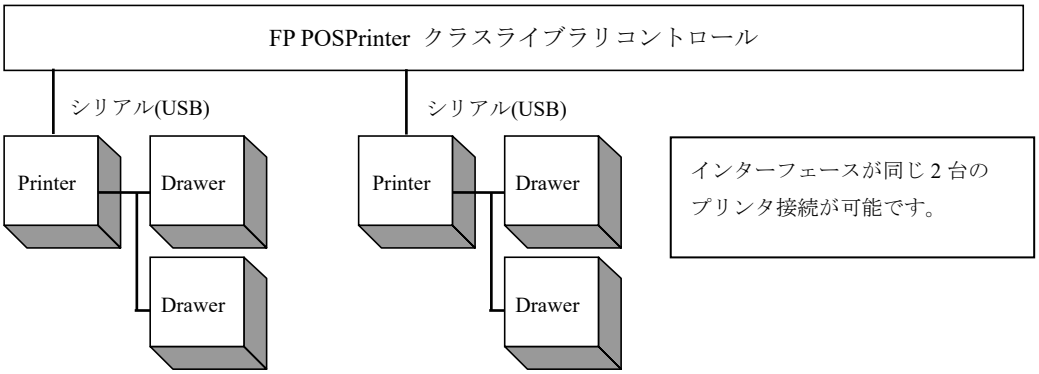
3. ログファイル単体のファイルサイズ上限はありません。詳細ログを出力している場合、POS(PC)のディスクスペースを圧迫するケースも考えられるので、注意してご使用ください。

4. **LogLevel** の設定によって下記ログファイルを出力します。
   LogLevel = 0 ： ログ出力なし
   LogLevel = 1 ： 通常トレースログ・警告・エラーログ出力
   LogLevel = 2 ： エラーログのみ出力

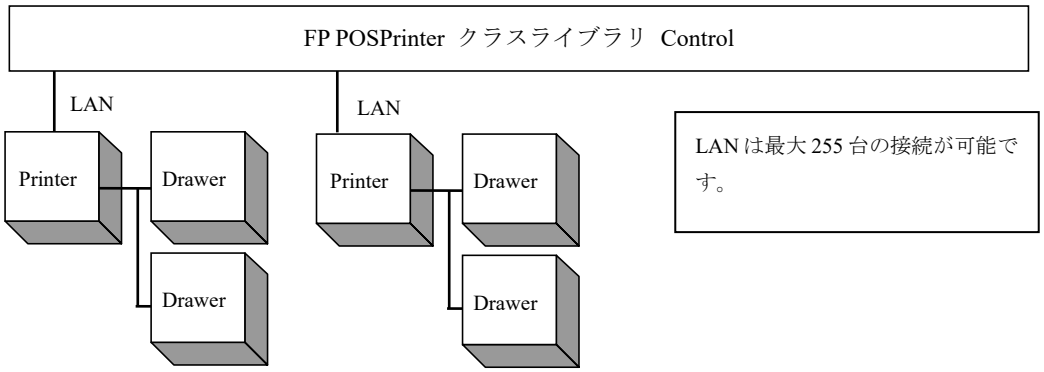
8. プリンター複数台の使用について

FP POSPrinter クラスライブラリコントロールは、シリアル・USB の 2 台の同時使用が可能となっています





FP POSPrinter クラスライブラリコントロールは、LAN インターフェースで最大 255 台の同時使用が可能となっています。



9. ページモード機能について

9.1. バッファリング機能を持つメソッドの排他関係について

PageModePrint メソッドと、同様のバッファリング機能を持つ TransactionPrint、RotatePrint メソッドの排他関係について、以下の表で示します。
「先発」を実行した後に「後発」を実行した場合、「○」は ErrorCode「例外なし」、「×」は ErrorCode「Illegal」を返却します。
「△」は ErrorCode「例外なし」となりますが、それまでのバッファリングデータは破棄されます。

			後発								
			PageModePrint			RotatePrint				Trasaction Print	
			PageMode	Normal	Cancel	Right90	Left90	Rotate180	Normal	Transaction	Normal
先発	未設定		○	×	×	○	○	○	○	○	○
	PageModePrint	PageMode	△	○	○	×	×	×	×	×	×
		Normal	○	×	×	○	○	○	○	○	○
		Cancel	○	×	×	○	○	○	○	○	○
	RotatePrint	Right90	×	×	×	×	×	×	○	×	×
		Left90	×	×	×	×	×	×	○	×	×
		Rotate180	×	×	×	×	×	×	○	×	×
		Normal	○	×	×	○	○	○	○	○	○
	TransactionPrint	Transaction	×	×	×	○	○	○	○	△	○
		Normal	○	×	×	○	○	○	○	○	○

（例） PageModePrint(PTR_PM_PAGE_MODE)実行後、TransactionPrint(Transaction)を実行した場合、ErrorCode「Illegal」を返却します。

9.2. ページモードでの印刷データ展開位置

ページモード開始中の絶対位置指定で、文字列、バーコード、及びビットマップ(PrintBitmap)はベースラインの下に印刷されます。

修正履歴			
文書名	FP-2000 シリーズ POSPrinter, CashDrawer シリアル・USB・LAN インターフェース対応 OPOS.NET クラスライブラリ アプリケーションプログラマーズガイド		
版数	変更日	変更箇所／変更理由／変更内容	変更者
1.0	2021/07/30	新規作成。	富士通アイソテック株式会社